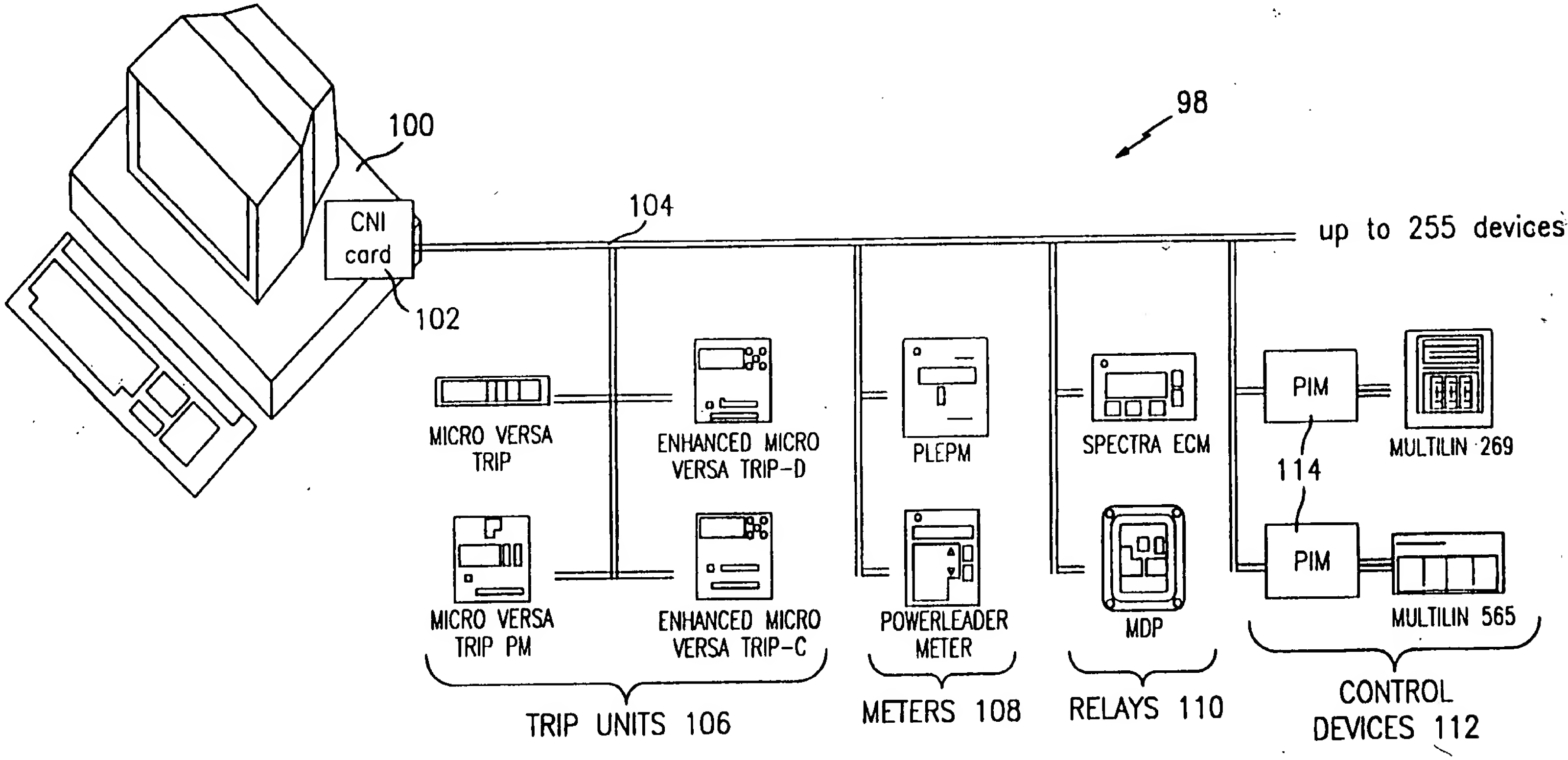


FIG. 1
(PRIOR ART)

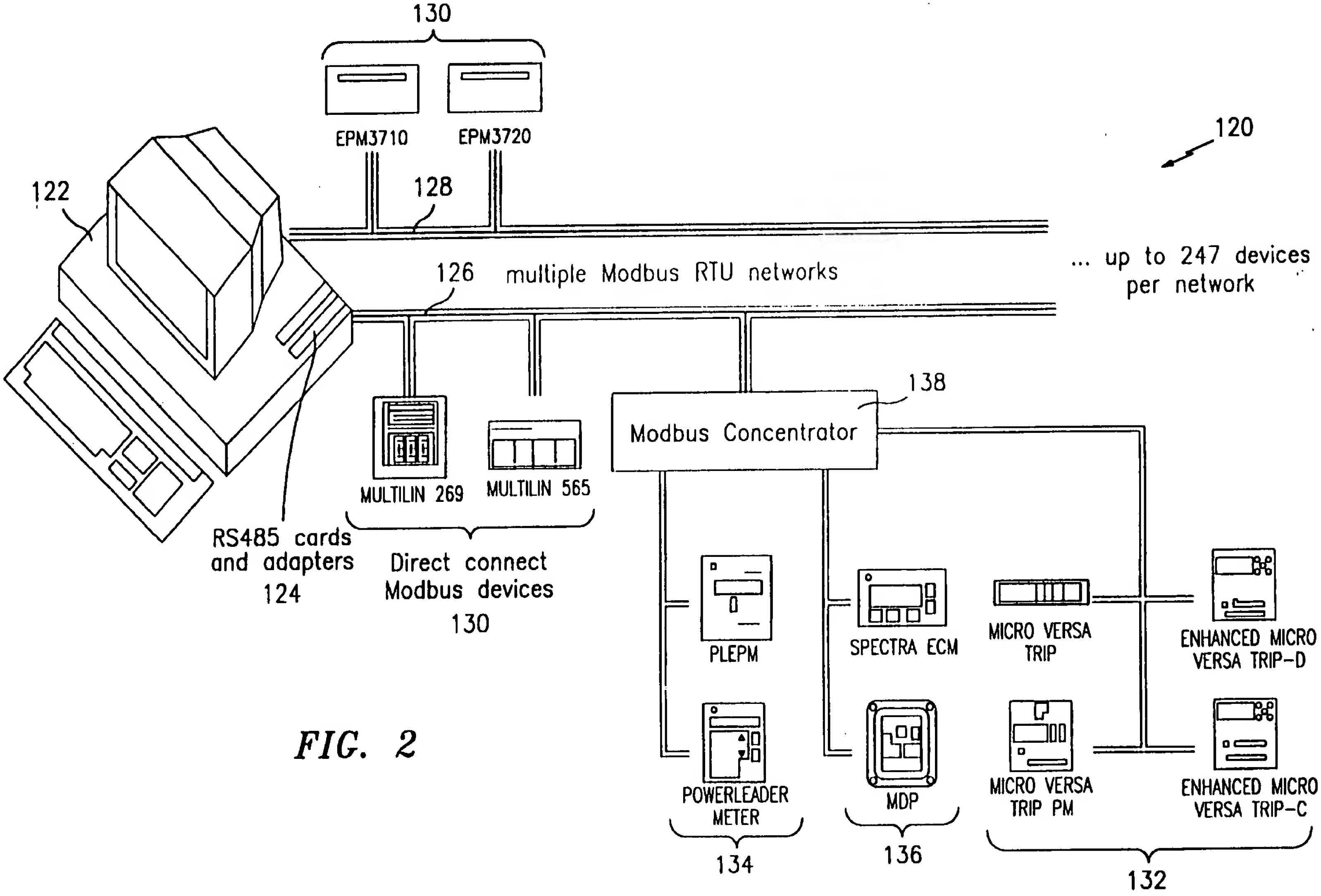

120
130
EPM3710
EPM3720
128
122
126
multiple Modbus RTU networks
... up to 247 devices per network
138
Modbus Concentrator
MULTILIN 269
MULTILIN 565
RS485 cards and adapters
124
Direct connect Modbus devices
130
PLEPM
SPECTRA ECM
MICRO VERSA TRIP
ENHANCED MICRO VERSA TRIP-D
POWERLEADER METER
MDP
MICRO VERSA TRIP PM
ENHANCED MICRO VERSA TRIP-C
134
136
132

FIG. 2

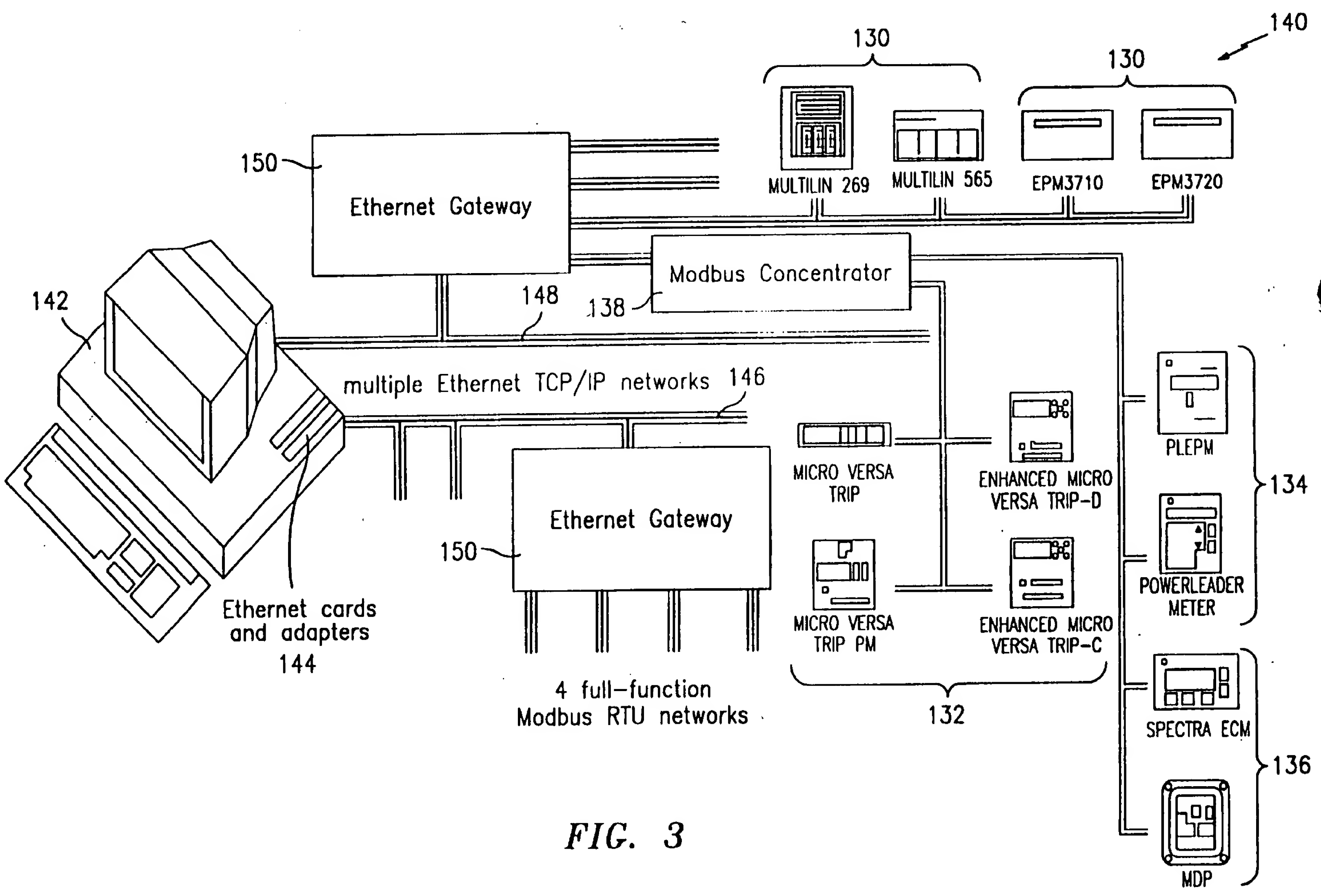
140
130
MULTILIN 269
MULTILIN 565
130
EPM3710
EPM3720
150
Ethernet Gateway
Modbus Concentrator
138
148
142
multiple Ethernet TCP/IP networks
146
MICRO VERSA TRIP
ENHANCED MICRO VERSA TRIP-D
MICRO VERSA TRIP PM
ENHANCED MICRO VERSA TRIP-C
132
PLEPM
POWERLEADER METER
134
SPECTRA ECM
MDP
136
150
Ethernet Gateway
Ethernet cards and adapters
144
4 full-function Modbus RTU networks

FIG. 3

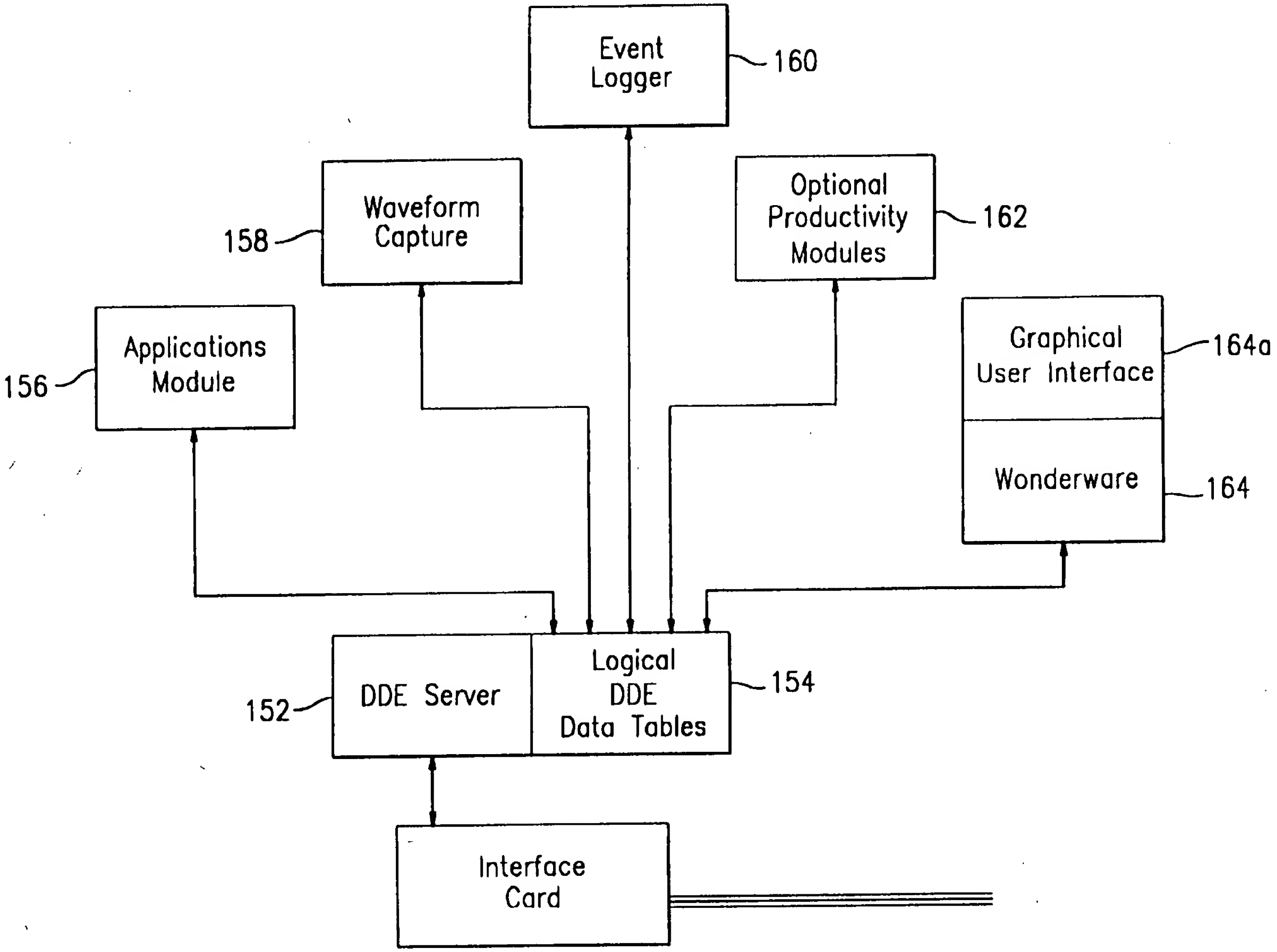
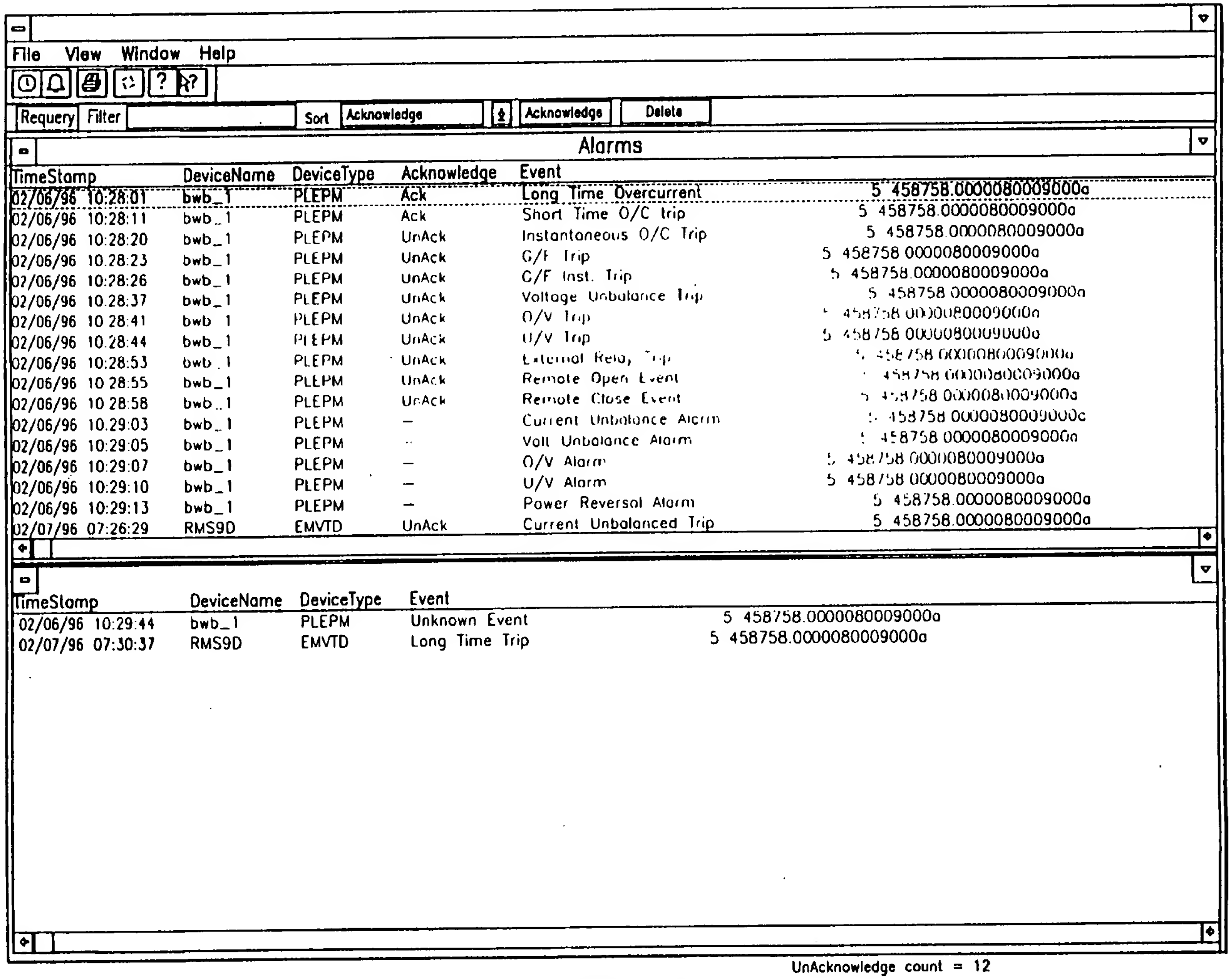
Event
Logger
160
Waveform
Capture
158
Optional
Productivity
Modules
162
Applications
Module
156
Graphical
User Interface
164a
Wonderware
164
DDE Server
152
Logical
DDE
Data Tables
154
Interface
Card

FIG. 4



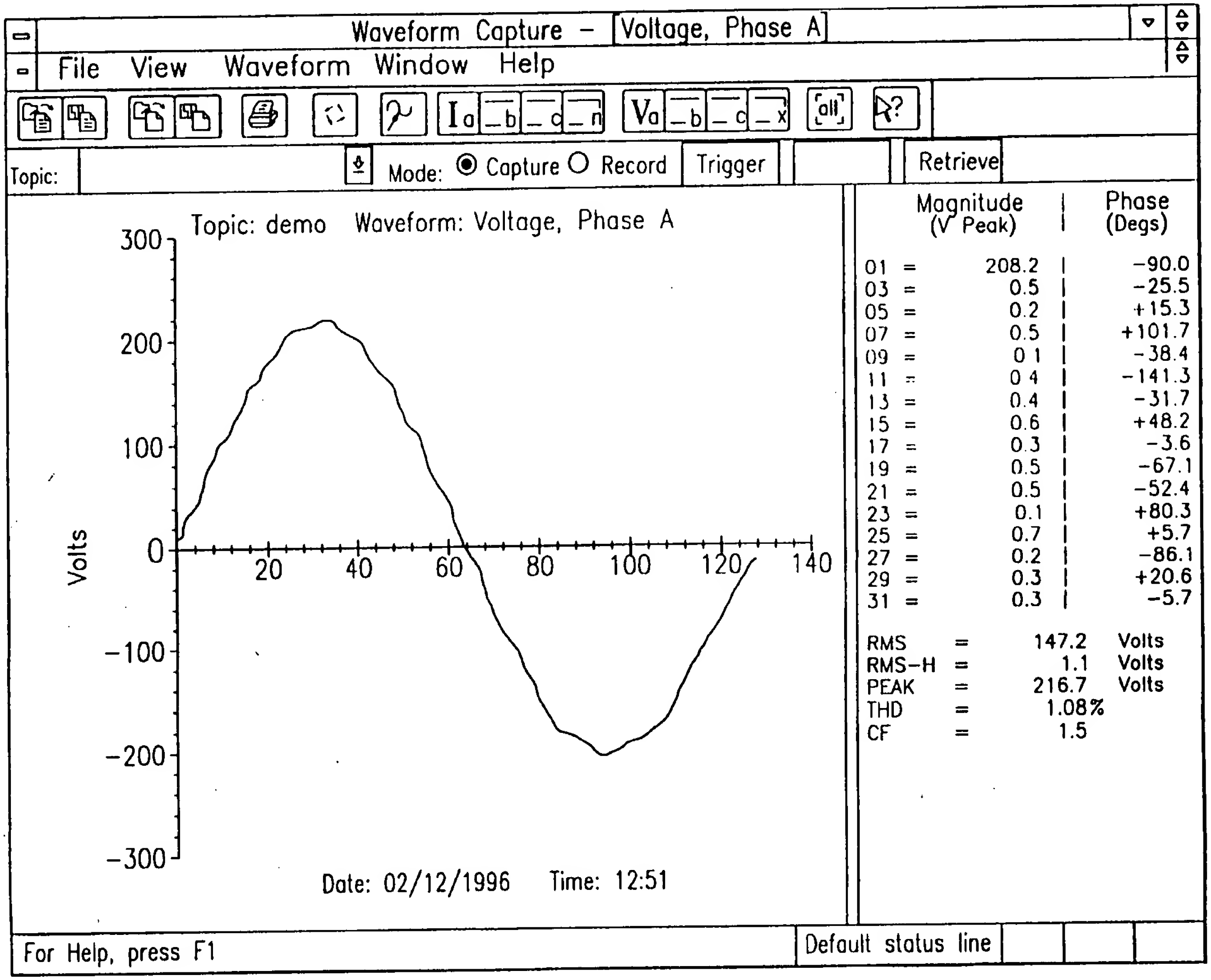
Alarms

TimeStamp	DeviceName	DeviceType	Acknowledge	Event	
02/06/96 10:28:01	bwb_1	PLEPM	Ack	Long Time Overcurrent	5 458758.0000080009000a
02/06/96 10:28:11	bwb_1	PLEPM	Ack	Short Time O/C trip	5 458758.0000080009000a
02/06/96 10:28:20	bwb_1	PLEPM	UnAck	Instantaneous O/C Trip	5 458758.0000080009000a
02/06/96 10:28:23	bwb_1	PLEPM	UnAck	G/F Trip	5 458758 0000080009000a
02/06/96 10:28:26	bwb_1	PLEPM	UnAck	G/F Inst. Trip	5 458758.0000080009000a
02/06/96 10:28:37	bwb_1	PLEPM	UnAck	Voltage Unbalance Trip	5 458758 0000080009000a
02/06/96 10:28:41	bwb 1	PLEPM	UnAck	O/V Trip	5 458758 0000080009000a
02/06/96 10:28:44	bwb_1	PLEPM	UnAck	U/V Trip	5 458758 0000080009000a
02/06/96 10:28:53	bwb_1	PLEPM	UnAck	External Relay Trip	5 458758 0000080009000a
02/06/96 10:28:55	bwb_1	PLEPM	UnAck	Remote Open Event	5 458758 0000080009000a
02/06/96 10:28:58	bwb_1	PLEPM	UnAck	Remote Close Event	5 458758 0000080009000a
02/06/96 10:29:03	bwb_1	PLEPM	—	Current Unbalance Alarm	5 458758 0000080009000c
02/06/96 10:29:05	bwb_1	PLEPM	—	Volt Unbalance Alarm	5 458758 0000080009000a
02/06/96 10:29:07	bwb_1	PLEPM	—	O/V Alarm	5 458758 0000080009000a
02/06/96 10:29:10	bwb_1	PLEPM	—	U/V Alarm	5 458758 0000080009000a
02/06/96 10:29:13	bwb_1	PLEPM	—	Power Reversal Alarm	5 458758.0000080009000a
02/07/96 07:26:29	RMS9D	EMVTD	UnAck	Current Unbalanced Trip	5 458758.0000080009000a

TimeStamp	DeviceName	DeviceType	Event	
02/06/96 10:29:44	bwb_1	PLEPM	Unknown Event	5 458758.0000080009000a
02/07/96 07:30:37	RMS9D	EMVTD	Long Time Trip	5 458758.0000080009000a

UnAcknowledge count = 12

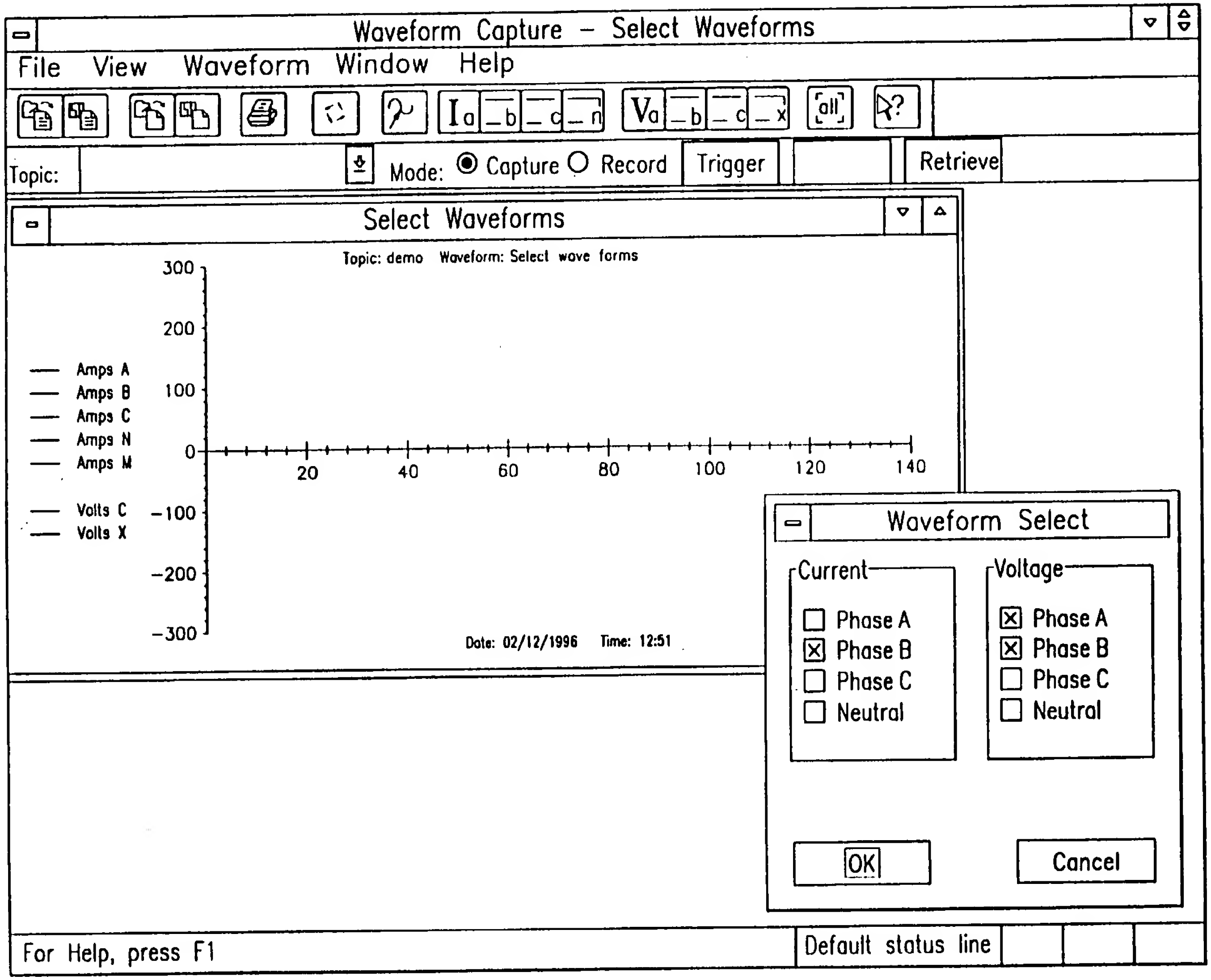
FIG. 5



	Magnitude (V Peak)	Phase (Degs)
01 =	208.2	-90.0
03 =	0.5	-25.5
05 =	0.2	+15.3
07 =	0.5	+101.7
09 =	0.1	-38.4
11 =	0.4	-141.3
13 =	0.4	-31.7
15 =	0.6	+48.2
17 =	0.3	-3.6
19 =	0.5	-67.1
21 =	0.5	-52.4
23 =	0.1	+80.3
25 =	0.7	+5.7
27 =	0.2	-86.1
29 =	0.3	+20.6
31 =	0.3	-5.7

RMS = 147.2 Volts
RMS-H = 1.1 Volts
PEAK = 216.7 Volts
THD = 1.08%
CF = 1.5

FIG. 6



Topic:

Mode: ● Capture ○ Record

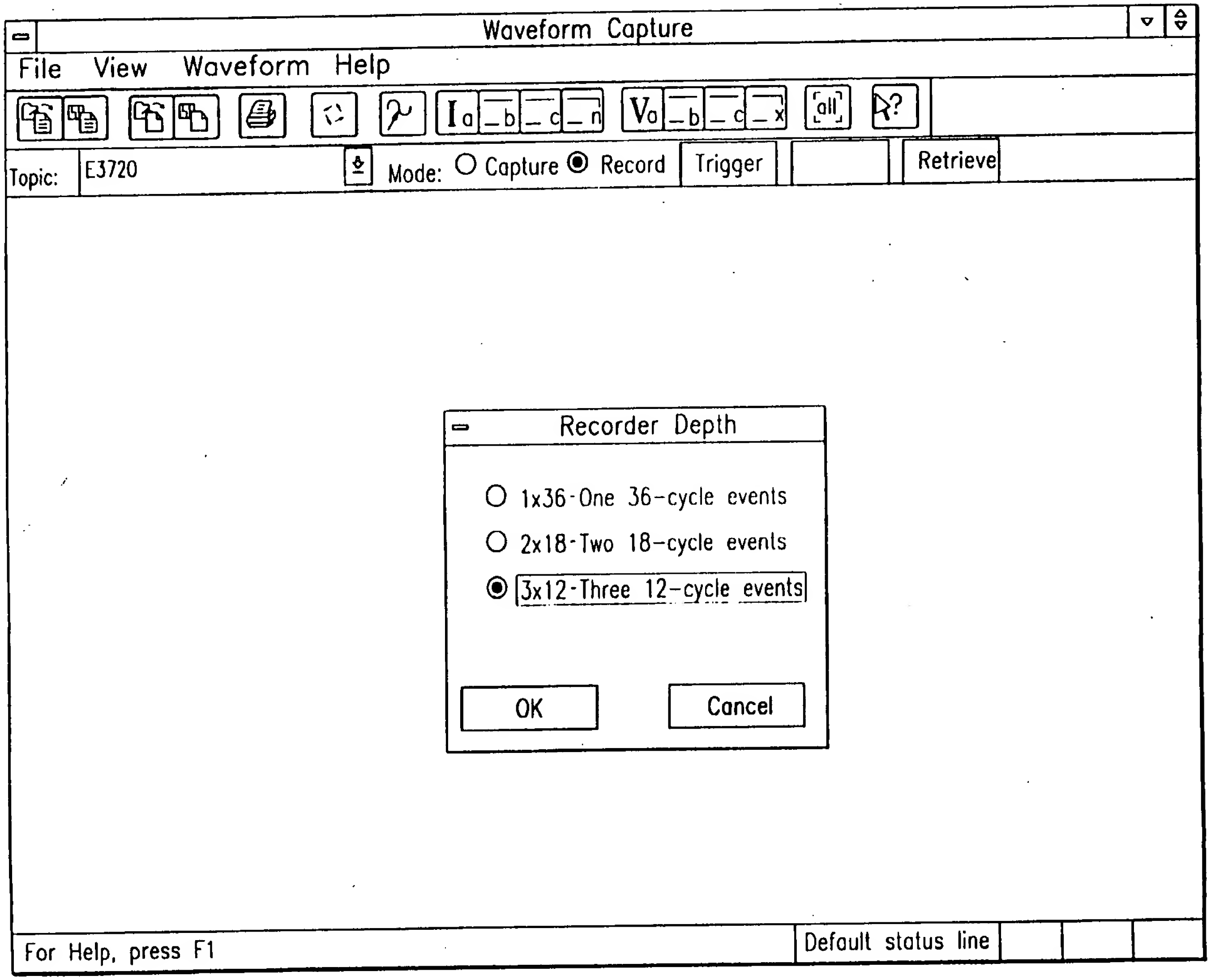
Topic: demo Waveform: Select wave forms

Date: 02/12/1996 Time: 12:51

Current
- ☐ Phase A
- ☒ Phase B
- ☐ Phase C
- ☐ Neutral

Voltage
- ☒ Phase A
- ☒ Phase B
- ☐ Phase C
- ☐ Neutral

FIG. 7



Topic: E3720

Mode: ○ Capture ● Record

Recorder Depth

○ 1x36-One 36-cycle events

○ 2x18-Two 18-cycle events

● 3x12-Three 12-cycle events

For Help, press F1

Default status line

FIG. 8

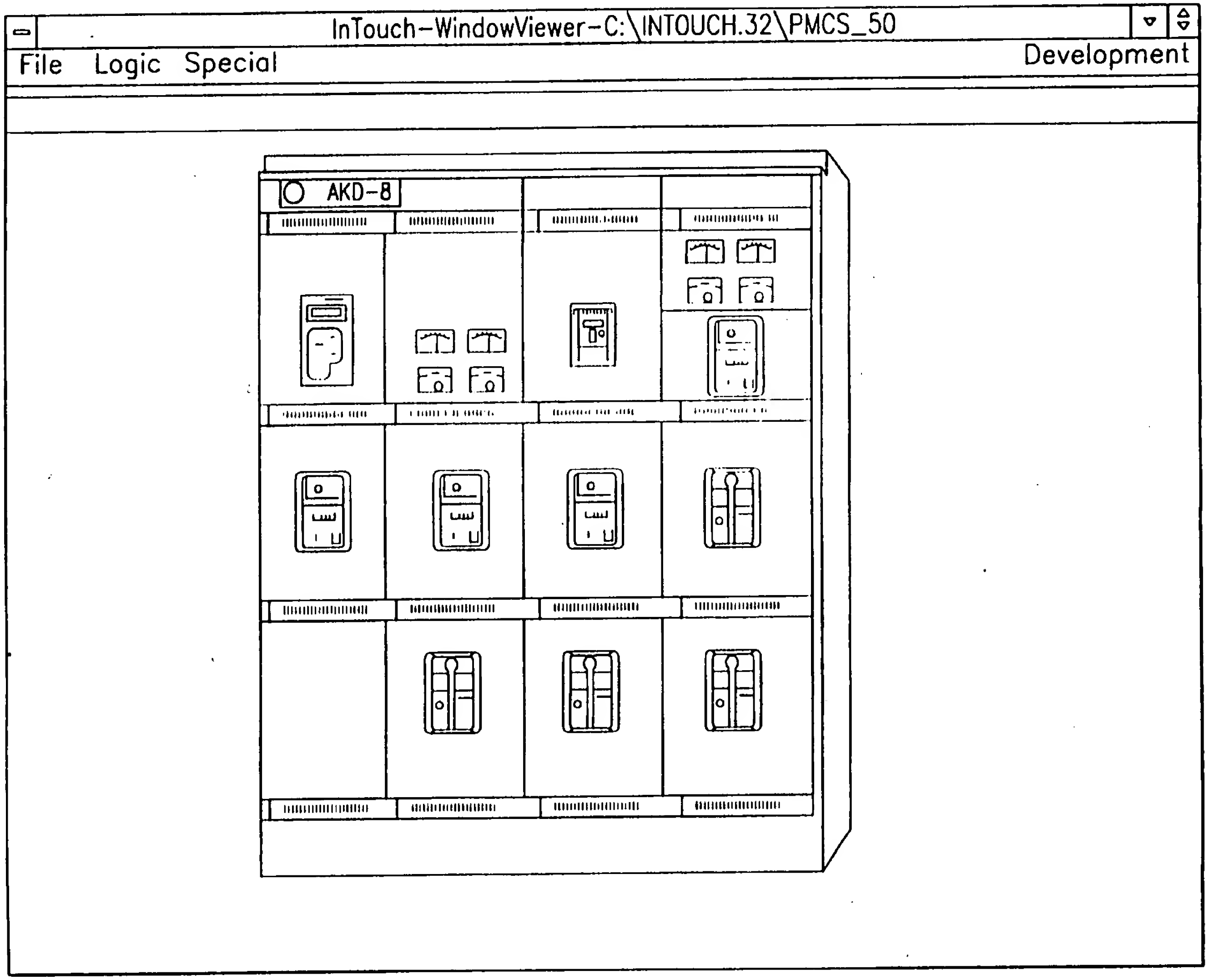
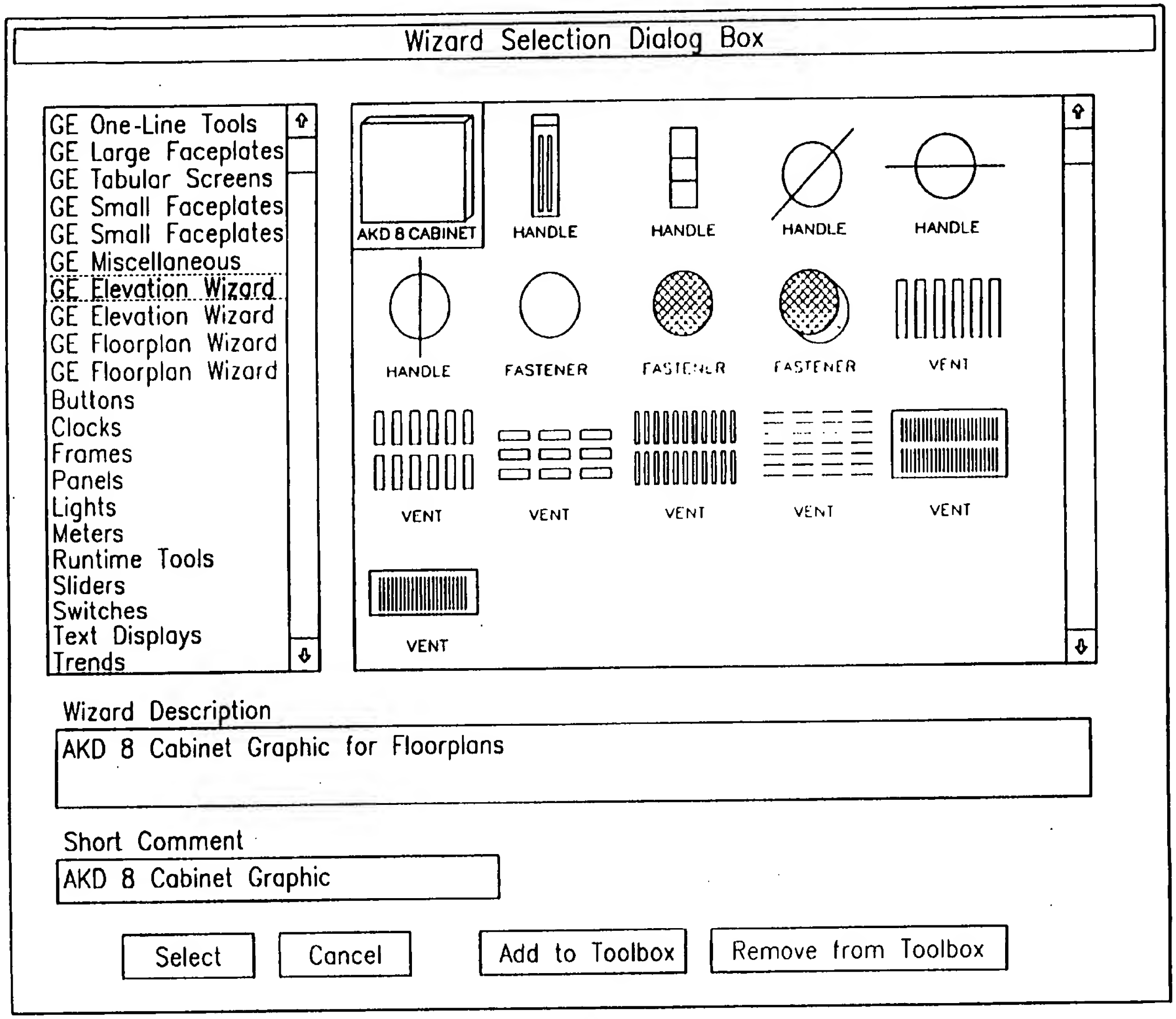

InTouch-WindowViewer-C:\INTOUCH.32\PMCS_50
File
Logic
Special
Development
AKD-8

FIG. 9



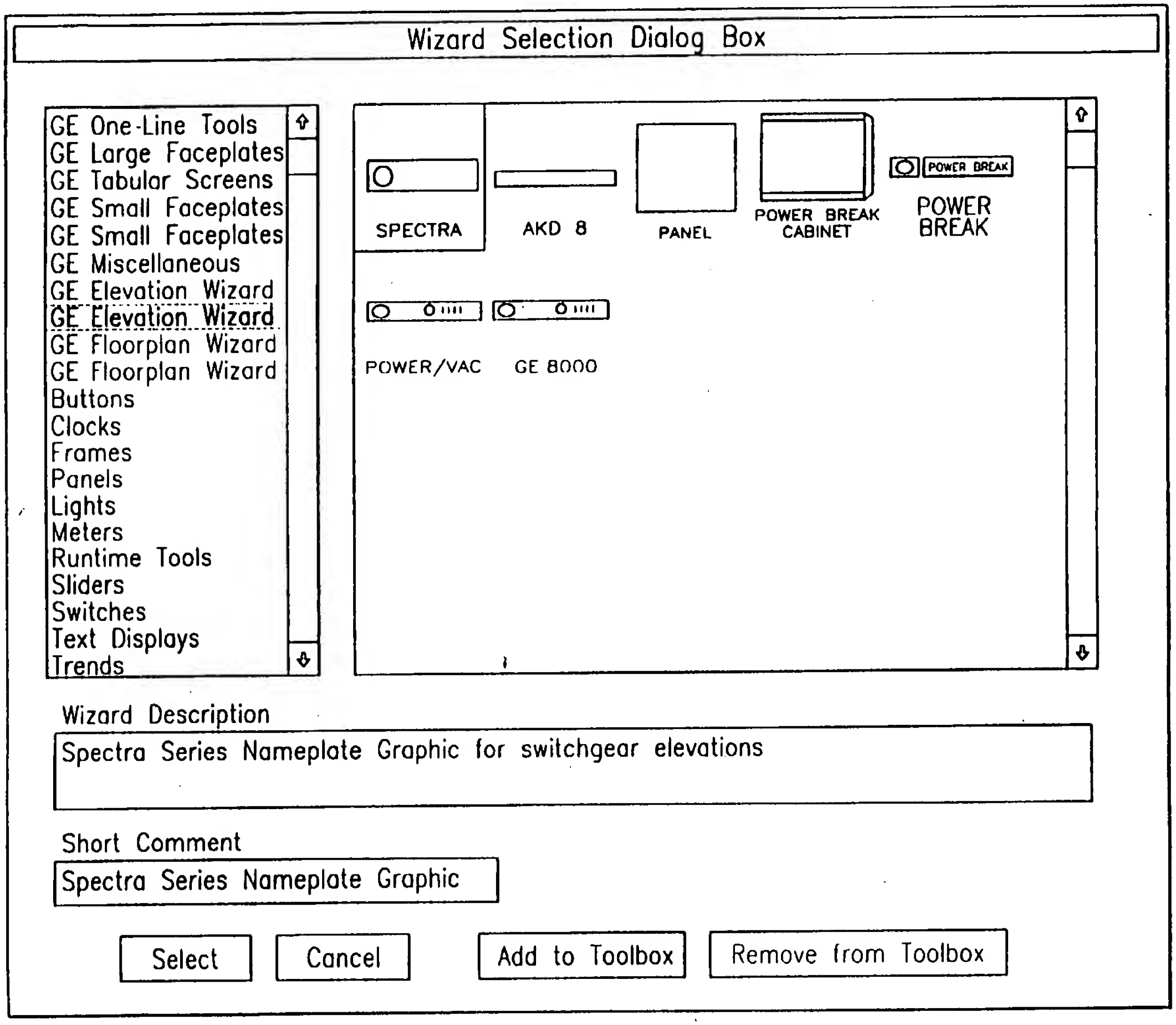
Wizard Description

AKD 8 Cabinet Graphic for Floorplans

Short Comment

AKD 8 Cabinet Graphic

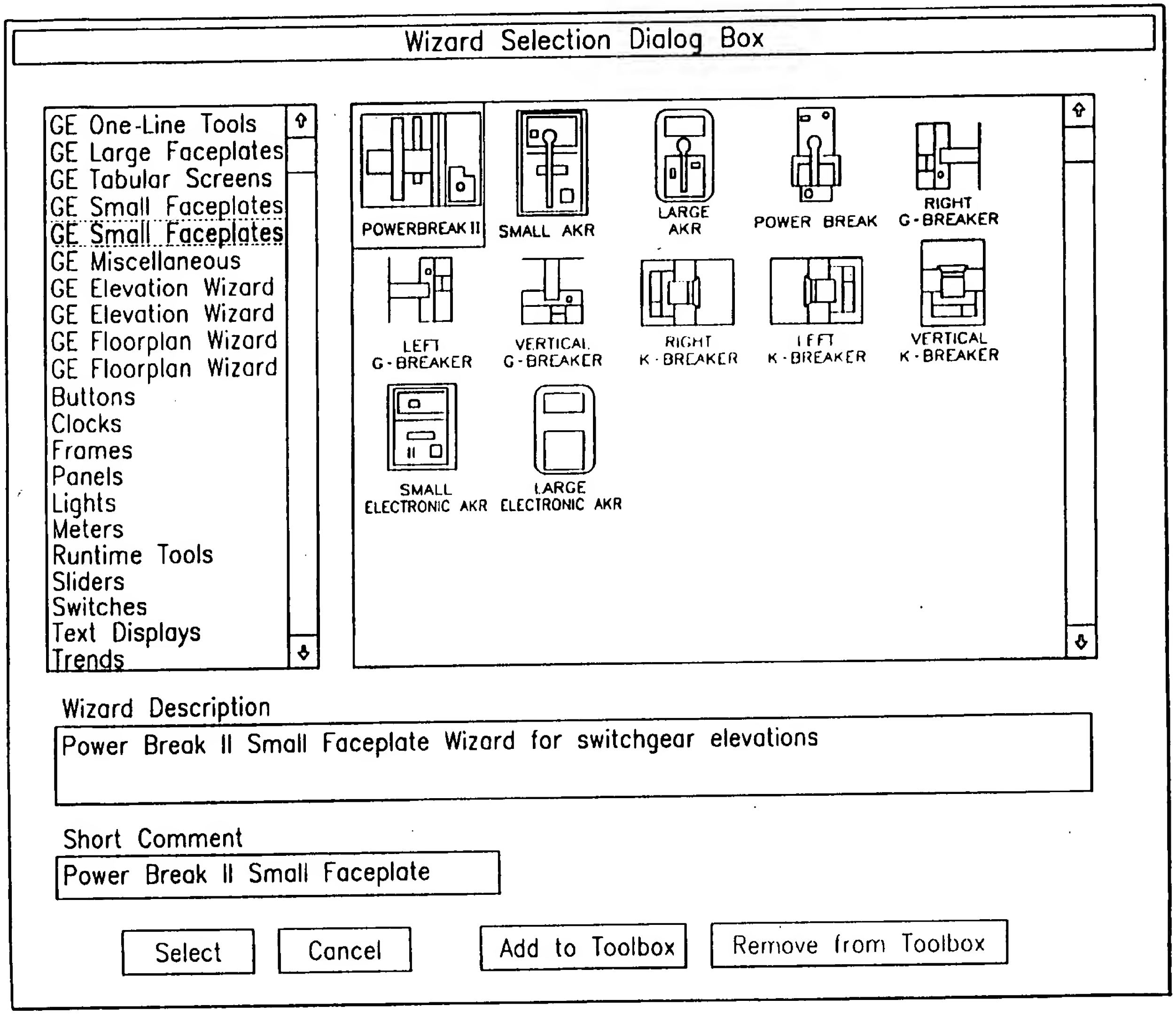
FIG. 10



Wizard Description
Spectra Series Nameplate Graphic for switchgear elevations

Short Comment
Spectra Series Nameplate Graphic

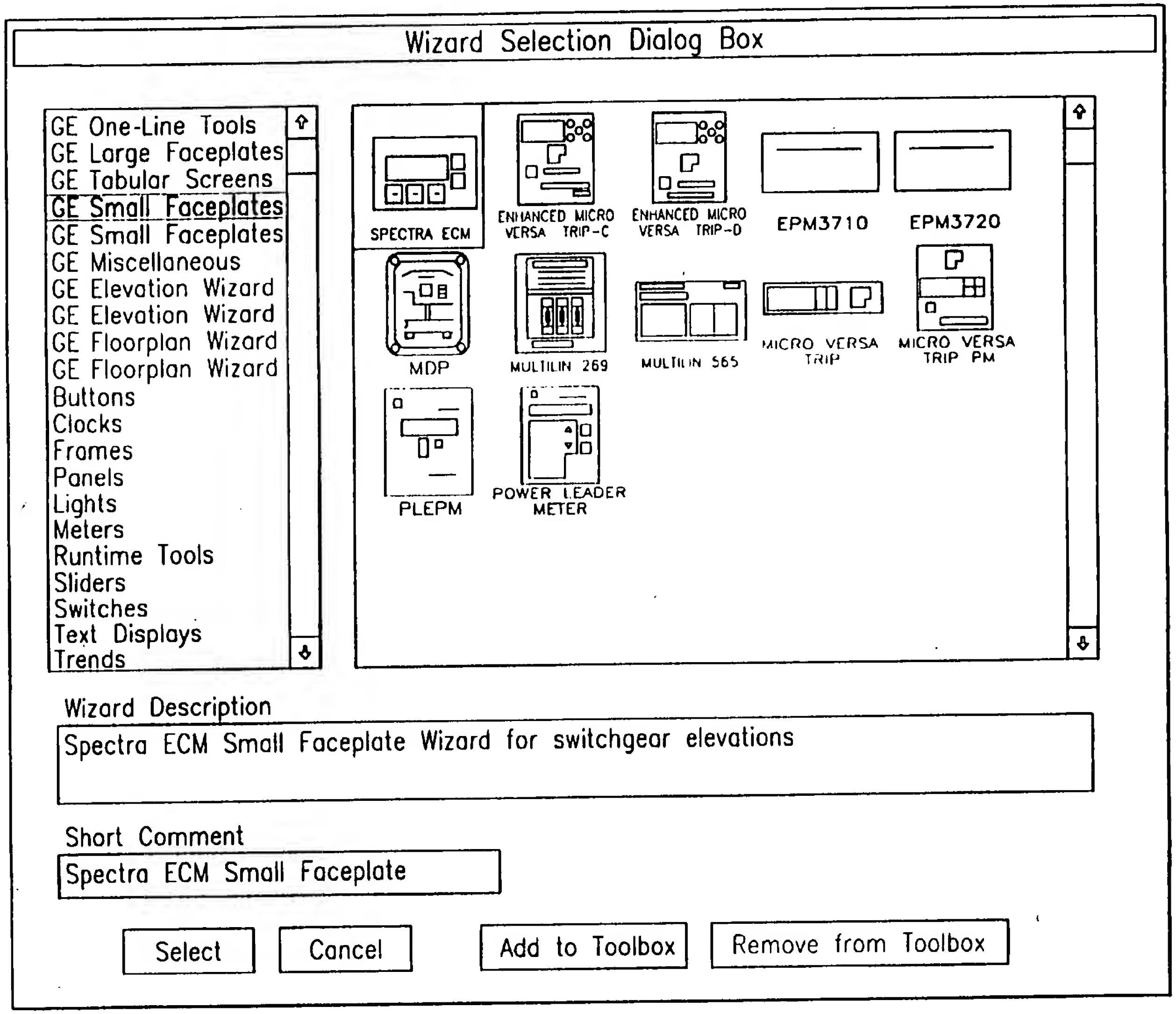
FIG. 11



Wizard Description: Power Break II Small Faceplate Wizard for switchgear elevations

Short Comment: Power Break II Small Faceplate

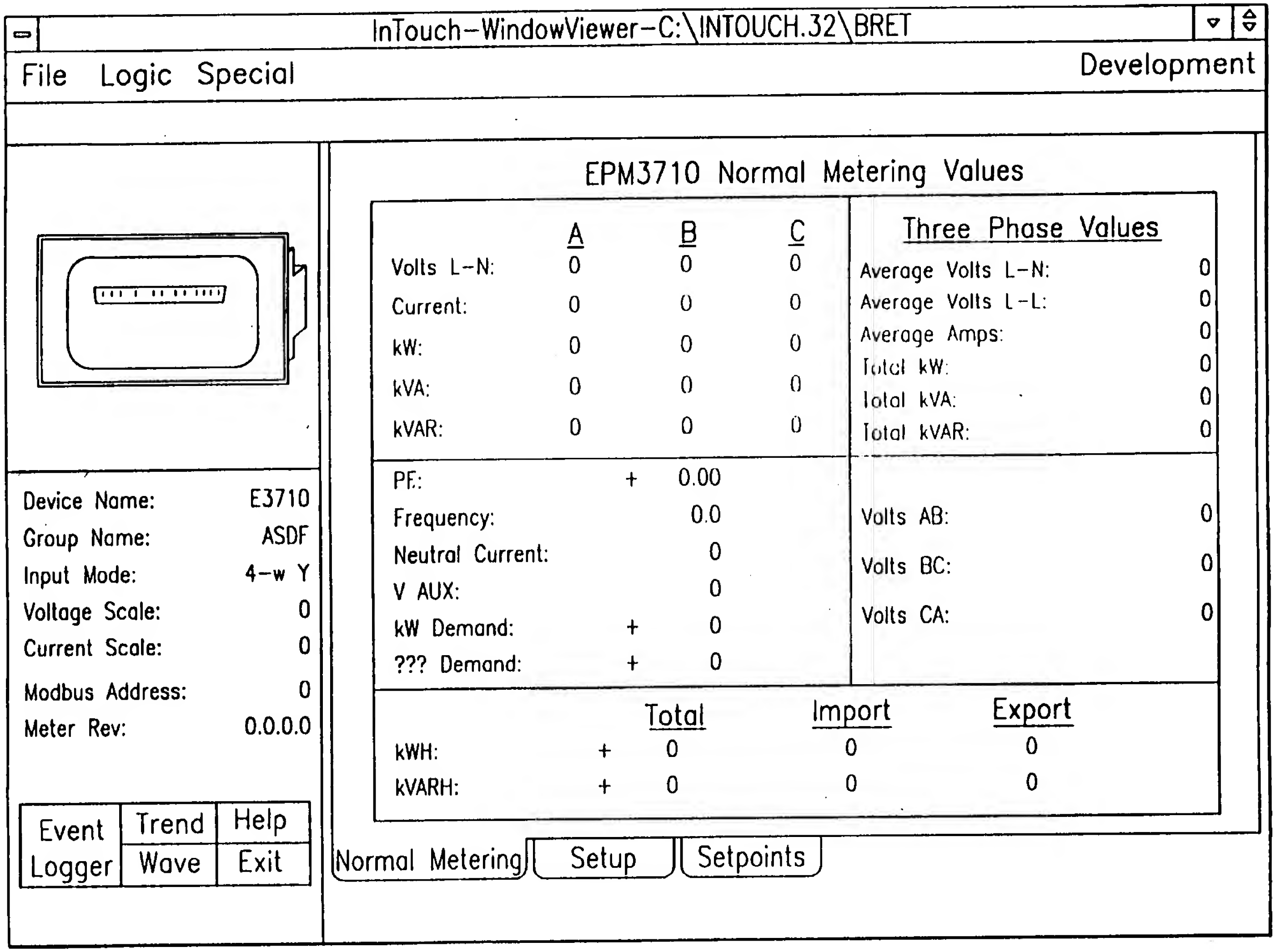
FIG. 12



Wizard Description: Spectra ECM Small Faceplate Wizard for switchgear elevations

Short Comment: Spectra ECM Small Faceplate

FIG. 13



Device Name:	E3710
Group Name:	ASDF
Input Mode:	4–w Y
Voltage Scale:	0
Current Scale:	0
Modbus Address:	0
Meter Rev:	0.0.0.0

EPM3710 Normal Metering Values

	A	B	C
Volts L–N:	0	0	0
Current:	0	0	0
kW:	0	0	0
kVA:	0	0	0
kVAR:	0	0	0

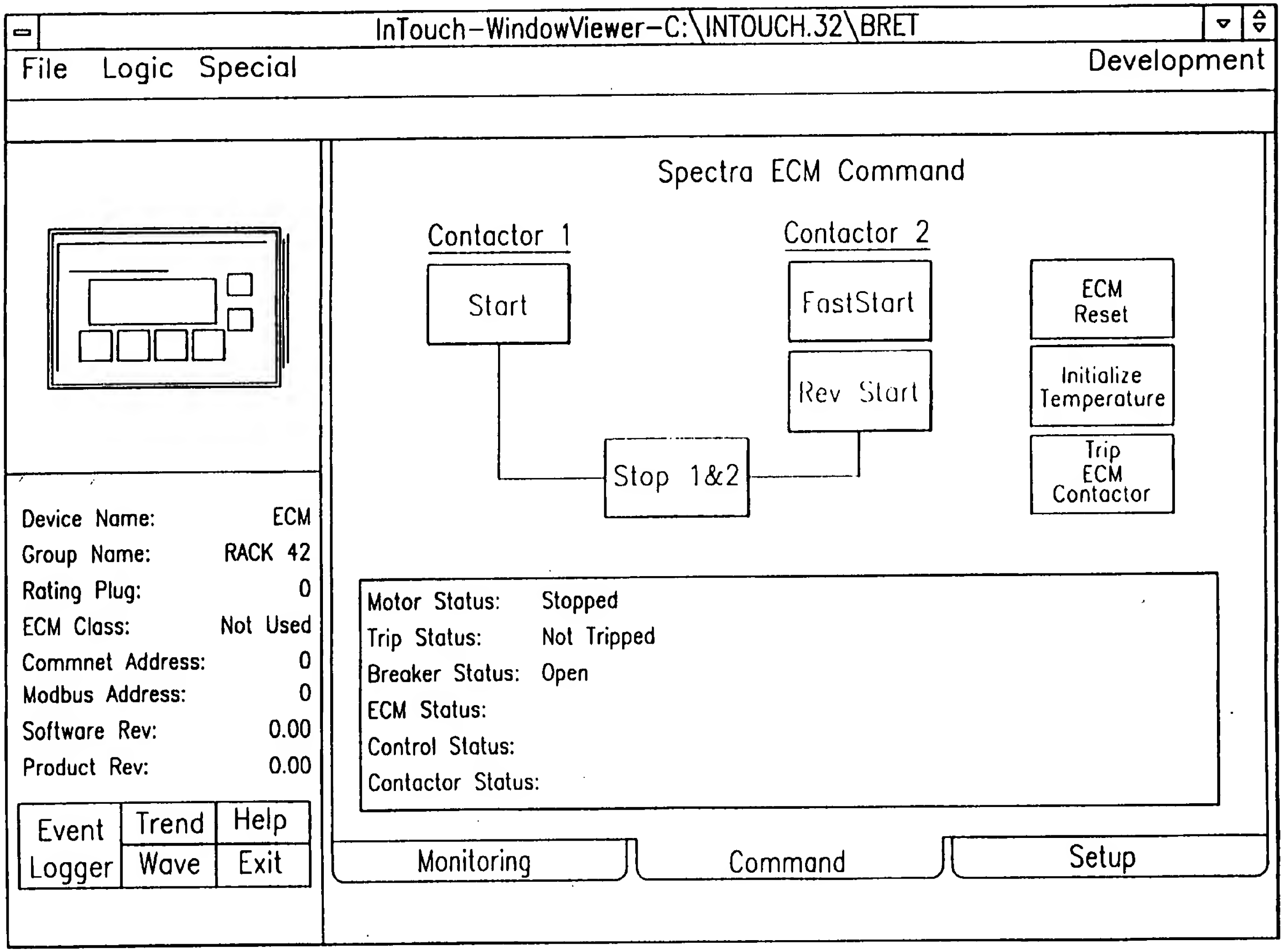
PF.:	+	0.00
Frequency:		0.0
Neutral Current:		0
V AUX:		0
kW Demand:	+	0
??? Demand:	+	0

Three Phase Values

Average Volts L–N:	0
Average Volts L–L:	0
Average Amps:	0
Total kW:	0
Total kVA:	0
Total kVAR:	0
Volts AB:	0
Volts BC:	0
Volts CA:	0

		Total	Import	Export
kWH:	+	0	0	0
kVARH:	+	0	0	0

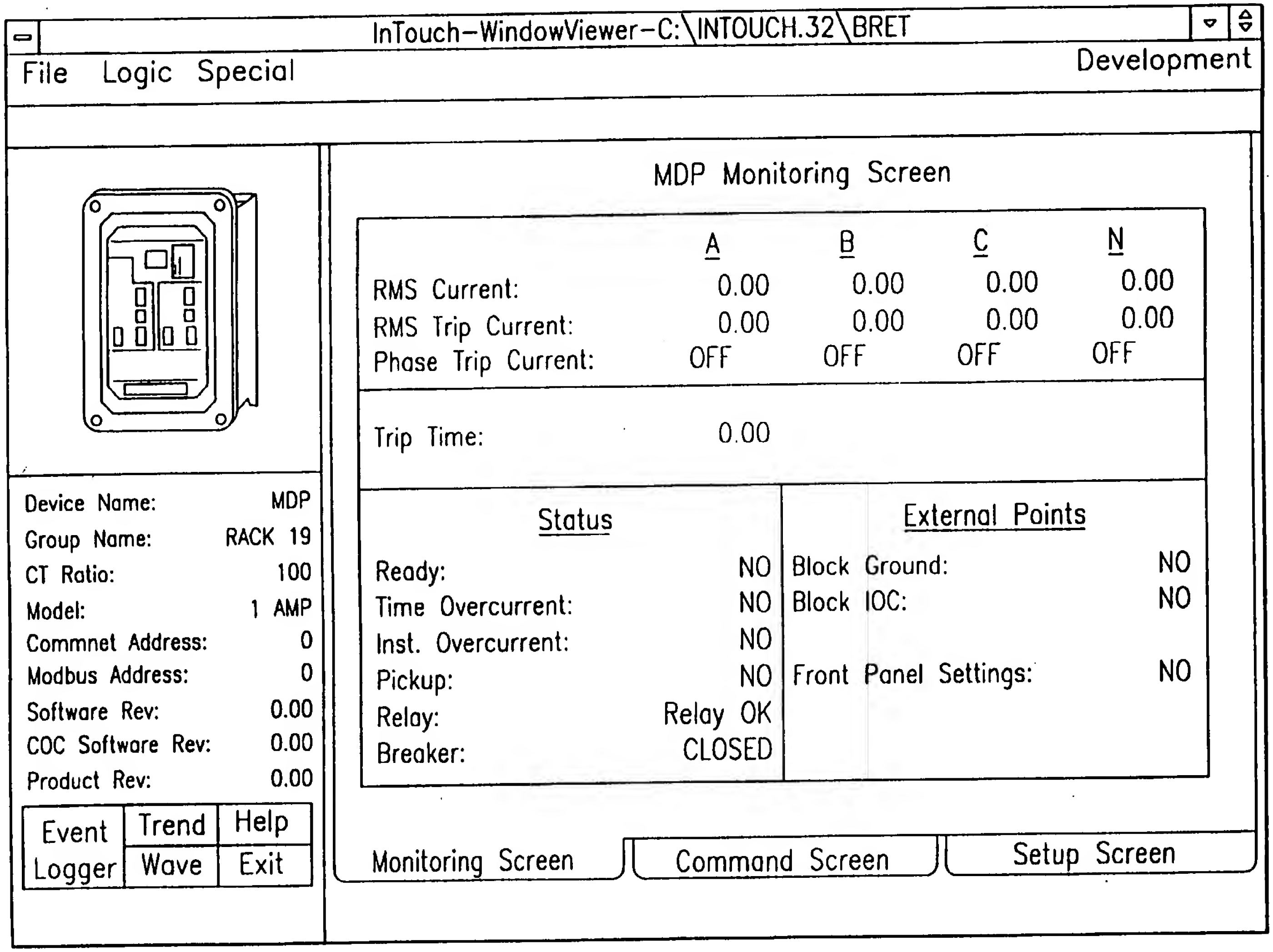
FIG. 14



Device Name: ECM
Group Name: RACK 42
Rating Plug: 0
ECM Class: Not Used
Commnet Address: 0
Modbus Address: 0
Software Rev: 0.00
Product Rev: 0.00

Motor Status: Stopped
Trip Status: Not Tripped
Breaker Status: Open
ECM Status:
Control Status:
Contactor Status:

FIG. 15



Device Name: MDP
Group Name: RACK 19
CT Ratio: 100
Model: 1 AMP
Commnet Address: 0
Modbus Address: 0
Software Rev: 0.00
COC Software Rev: 0.00
Product Rev: 0.00

MDP Monitoring Screen

	A	B	C	N
RMS Current:	0.00	0.00	0.00	0.00
RMS Trip Current:	0.00	0.00	0.00	0.00
Phase Trip Current:	OFF	OFF	OFF	OFF

Trip Time: 0.00

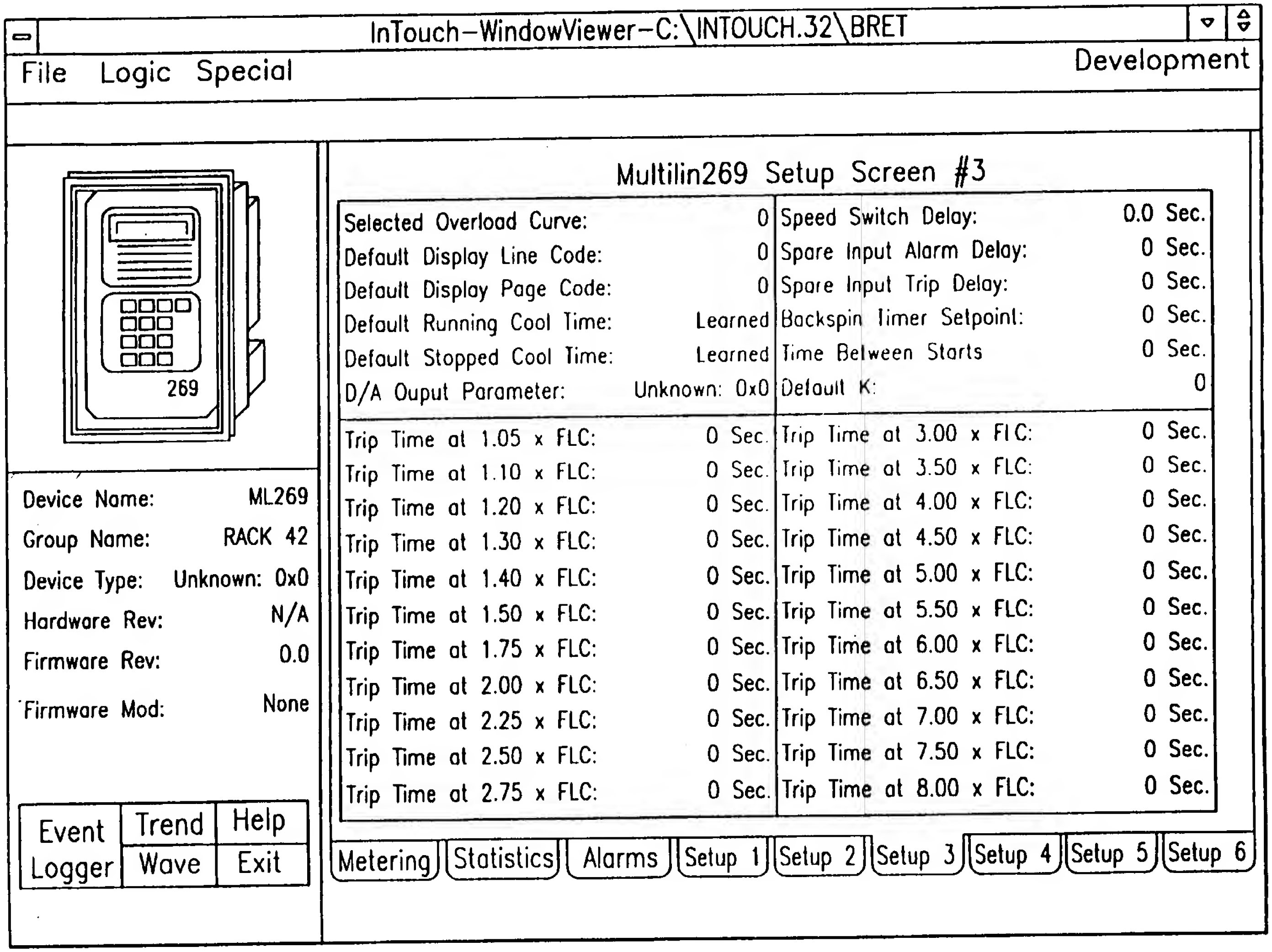
Status

Ready: NO
Time Overcurrent: NO
Inst. Overcurrent: NO
Pickup: NO
Relay: Relay OK
Breaker: CLOSED

External Points

Block Ground: NO
Block IOC: NO
Front Panel Settings: NO

FIG. 16

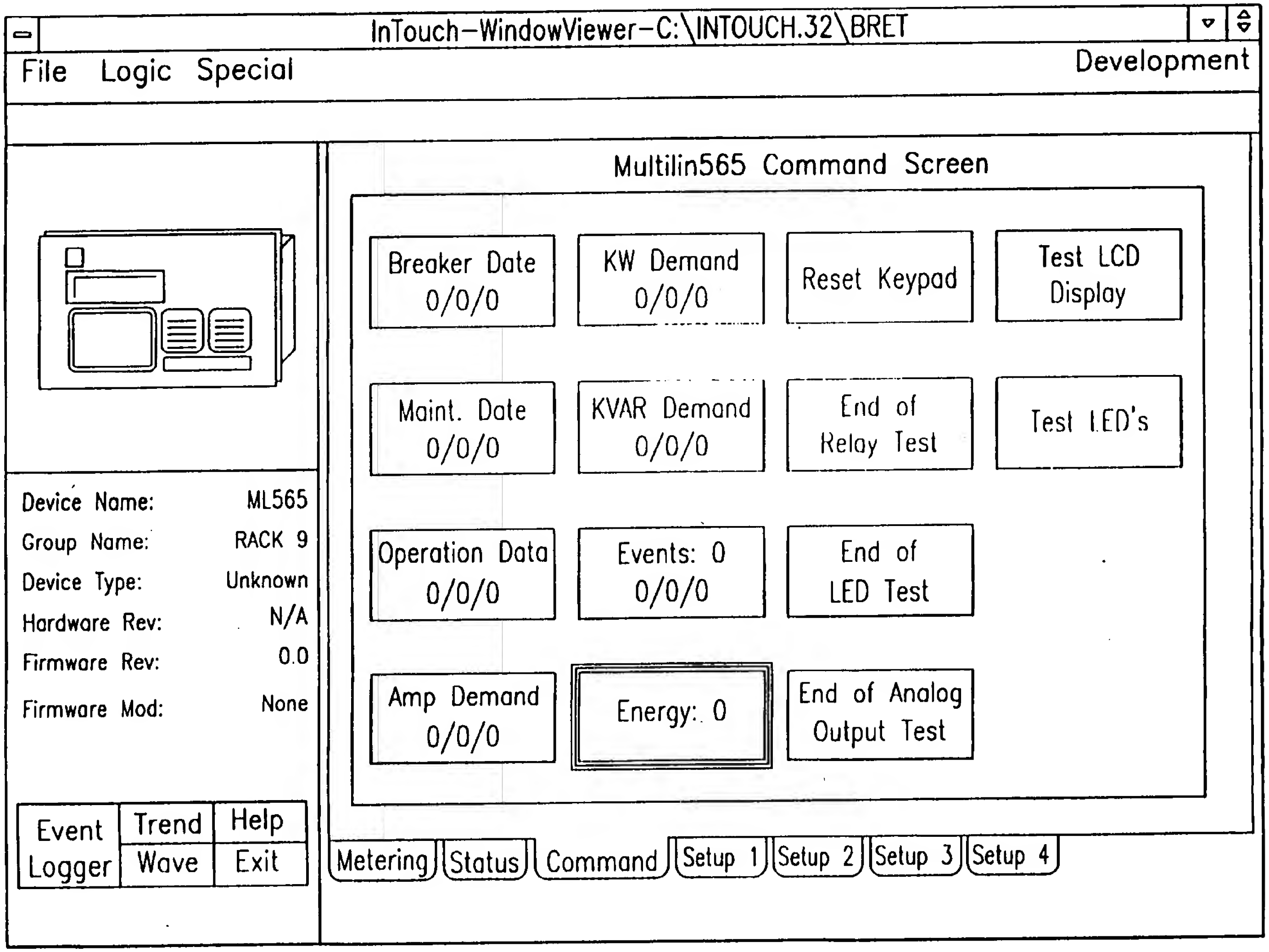


Device Name: ML269
Group Name: RACK 42
Device Type: Unknown: 0x0
Hardware Rev: N/A
Firmware Rev: 0.0
Firmware Mod: None

Multilin269 Setup Screen #3

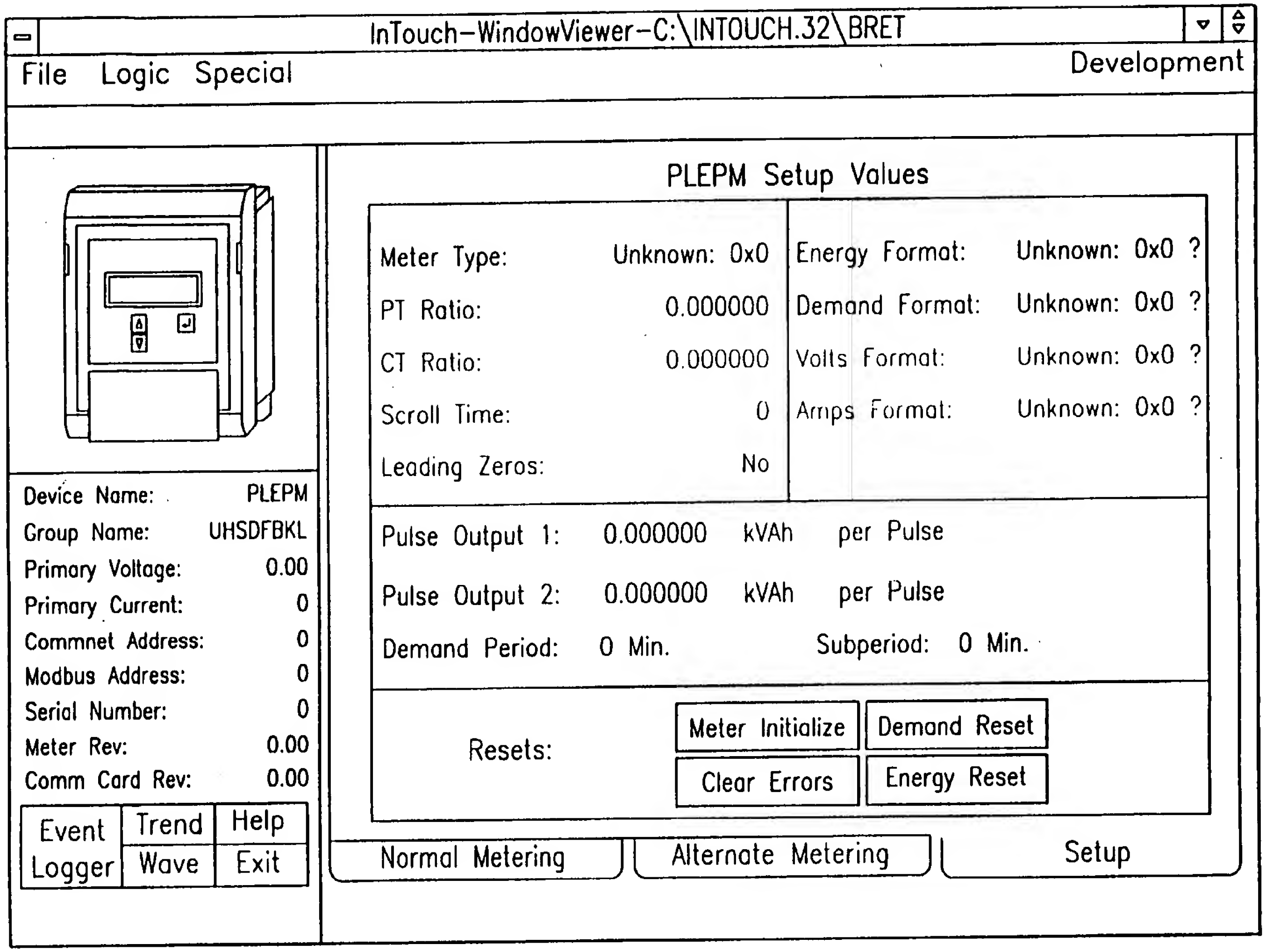
Selected Overload Curve:	0	Speed Switch Delay:	0.0 Sec.
Default Display Line Code:	0	Spare Input Alarm Delay:	0 Sec.
Default Display Page Code:	0	Spare Input Trip Delay:	0 Sec.
Default Running Cool Time:	Learned	Backspin Timer Setpoint:	0 Sec.
Default Stopped Cool Time:	Learned	Time Between Starts	0 Sec.
D/A Ouput Parameter:	Unknown: 0x0	Default K:	0
Trip Time at 1.05 x FLC:	0 Sec.	Trip Time at 3.00 x FLC:	0 Sec.
Trip Time at 1.10 x FLC:	0 Sec.	Trip Time at 3.50 x FLC:	0 Sec.
Trip Time at 1.20 x FLC:	0 Sec.	Trip Time at 4.00 x FLC:	0 Sec.
Trip Time at 1.30 x FLC:	0 Sec.	Trip Time at 4.50 x FLC:	0 Sec.
Trip Time at 1.40 x FLC:	0 Sec.	Trip Time at 5.00 x FLC:	0 Sec.
Trip Time at 1.50 x FLC:	0 Sec.	Trip Time at 5.50 x FLC:	0 Sec.
Trip Time at 1.75 x FLC:	0 Sec.	Trip Time at 6.00 x FLC:	0 Sec.
Trip Time at 2.00 x FLC:	0 Sec.	Trip Time at 6.50 x FLC:	0 Sec.
Trip Time at 2.25 x FLC:	0 Sec.	Trip Time at 7.00 x FLC:	0 Sec.
Trip Time at 2.50 x FLC:	0 Sec.	Trip Time at 7.50 x FLC:	0 Sec.
Trip Time at 2.75 x FLC:	0 Sec.	Trip Time at 8.00 x FLC:	0 Sec.

FIG. 17



Device Name: ML565
Group Name: RACK 9
Device Type: Unknown
Hardware Rev: N/A
Firmware Rev: 0.0
Firmware Mod: None

FIG. 18



Device Name: PLEPM
Group Name: UHSDFBKL
Primary Voltage: 0.00
Primary Current: 0
Commnet Address: 0
Modbus Address: 0
Serial Number: 0
Meter Rev: 0.00
Comm Card Rev: 0.00

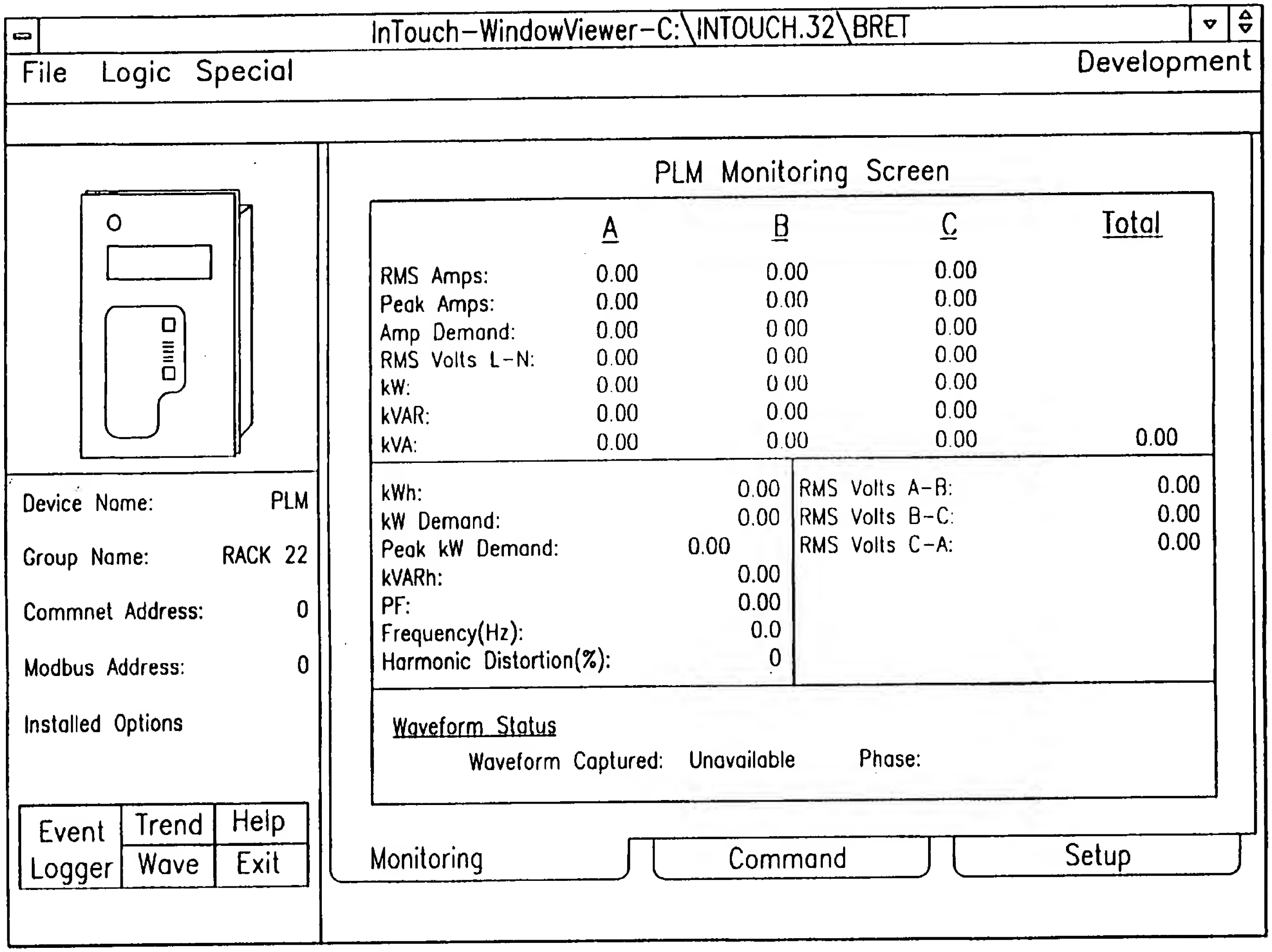
PLEPM Setup Values

Meter Type:	Unknown: 0x0	Energy Format:	Unknown: 0x0 ?
PT Ratio:	0.000000	Demand Format:	Unknown: 0x0 ?
CT Ratio:	0.000000	Volts Format:	Unknown: 0x0 ?
Scroll Time:	0	Amps Format:	Unknown: 0x0 ?
Leading Zeros:	No		

Pulse Output 1: 0.000000 kVAh per Pulse
Pulse Output 2: 0.000000 kVAh per Pulse
Demand Period: 0 Min. Subperiod: 0 Min.

Resets:

FIG. 19



Device Name: PLM

Group Name: RACK 22

Commnet Address: 0

Modbus Address: 0

Installed Options

PLM Monitoring Screen

	A	B	C	Total
RMS Amps:	0.00	0.00	0.00	
Peak Amps:	0.00	0.00	0.00	
Amp Demand:	0.00	0.00	0.00	
RMS Volts L-N:	0.00	0.00	0.00	
kW:	0.00	0.00	0.00	
kVAR:	0.00	0.00	0.00	
kVA:	0.00	0.00	0.00	0.00

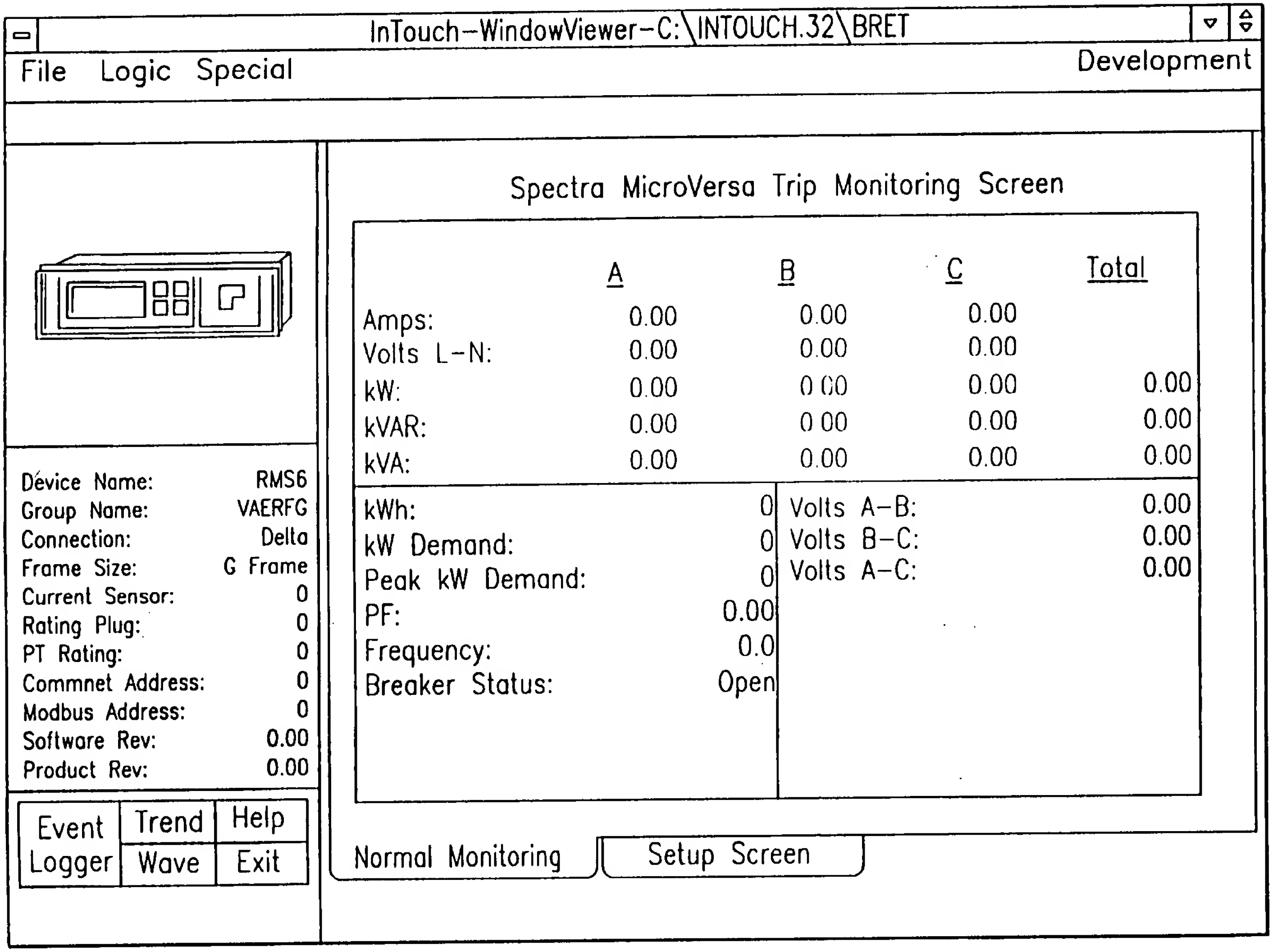
kWh:	0.00
kW Demand:	0.00
Peak kW Demand:	0.00
kVARh:	0.00
PF:	0.00
Frequency(Hz):	0.0
Harmonic Distortion(%):	0

RMS Volts A-B:	0.00
RMS Volts B-C:	0.00
RMS Volts C-A:	0.00

Waveform Status

Waveform Captured: Unavailable Phase:

FIG. 20



Device Name: RMS6
Group Name: VAERFG
Connection: Delta
Frame Size: G Frame
Current Sensor: 0
Rating Plug: 0
PT Rating: 0
Commnet Address: 0
Modbus Address: 0
Software Rev: 0.00
Product Rev: 0.00

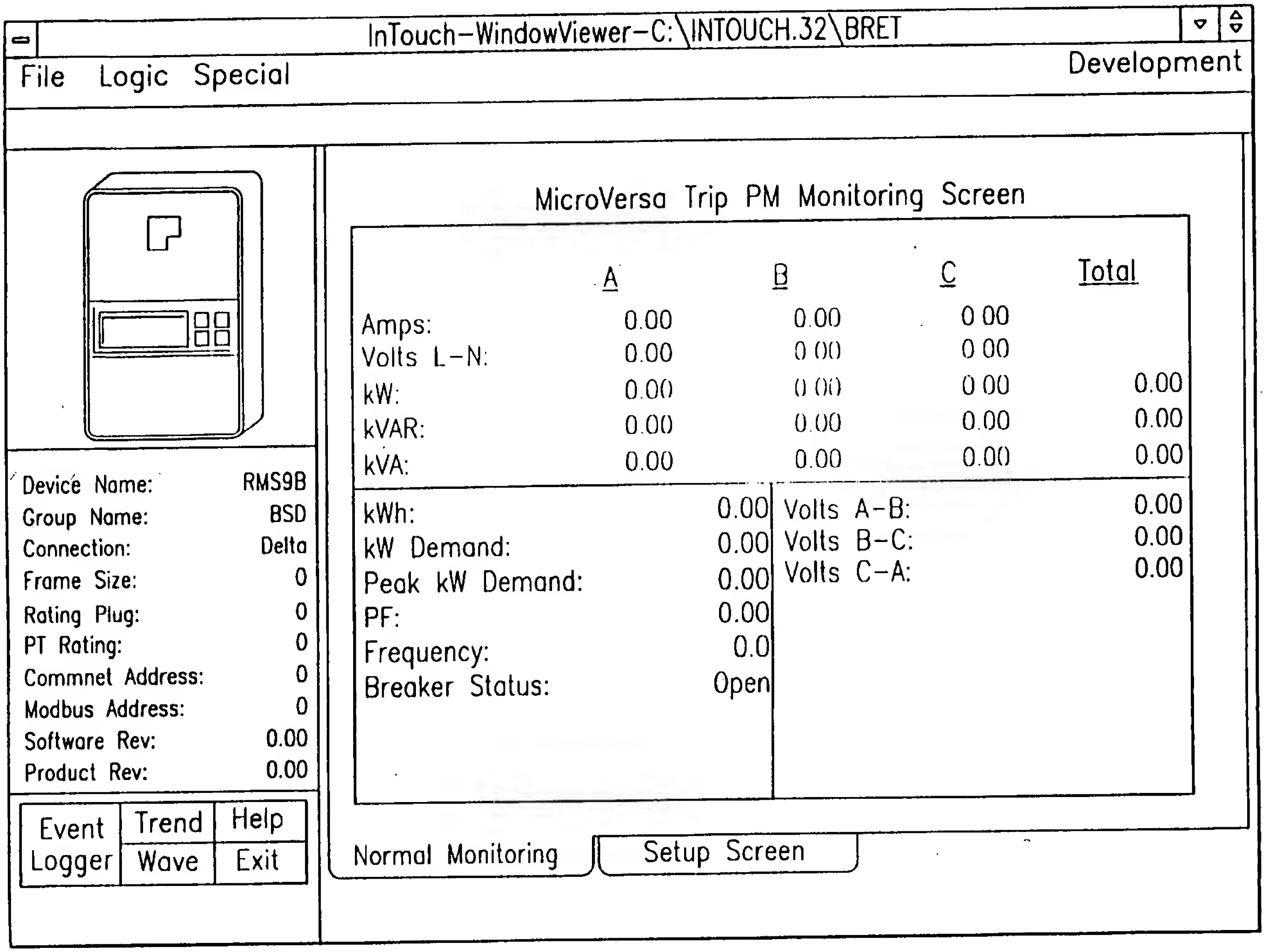
Spectra MicroVersa Trip Monitoring Screen

	A	B	C	Total
Amps:	0.00	0.00	0.00	
Volts L–N:	0.00	0.00	0.00	
kW:	0.00	0.00	0.00	0.00
kVAR:	0.00	0.00	0.00	0.00
kVA:	0.00	0.00	0.00	0.00

kWh: 0
kW Demand: 0
Peak kW Demand: 0
PF: 0.00
Frequency: 0.0
Breaker Status: Open

Volts A–B: 0.00
Volts B–C: 0.00
Volts A–C: 0.00

FIG. 21



MicroVersa Trip PM Monitoring Screen

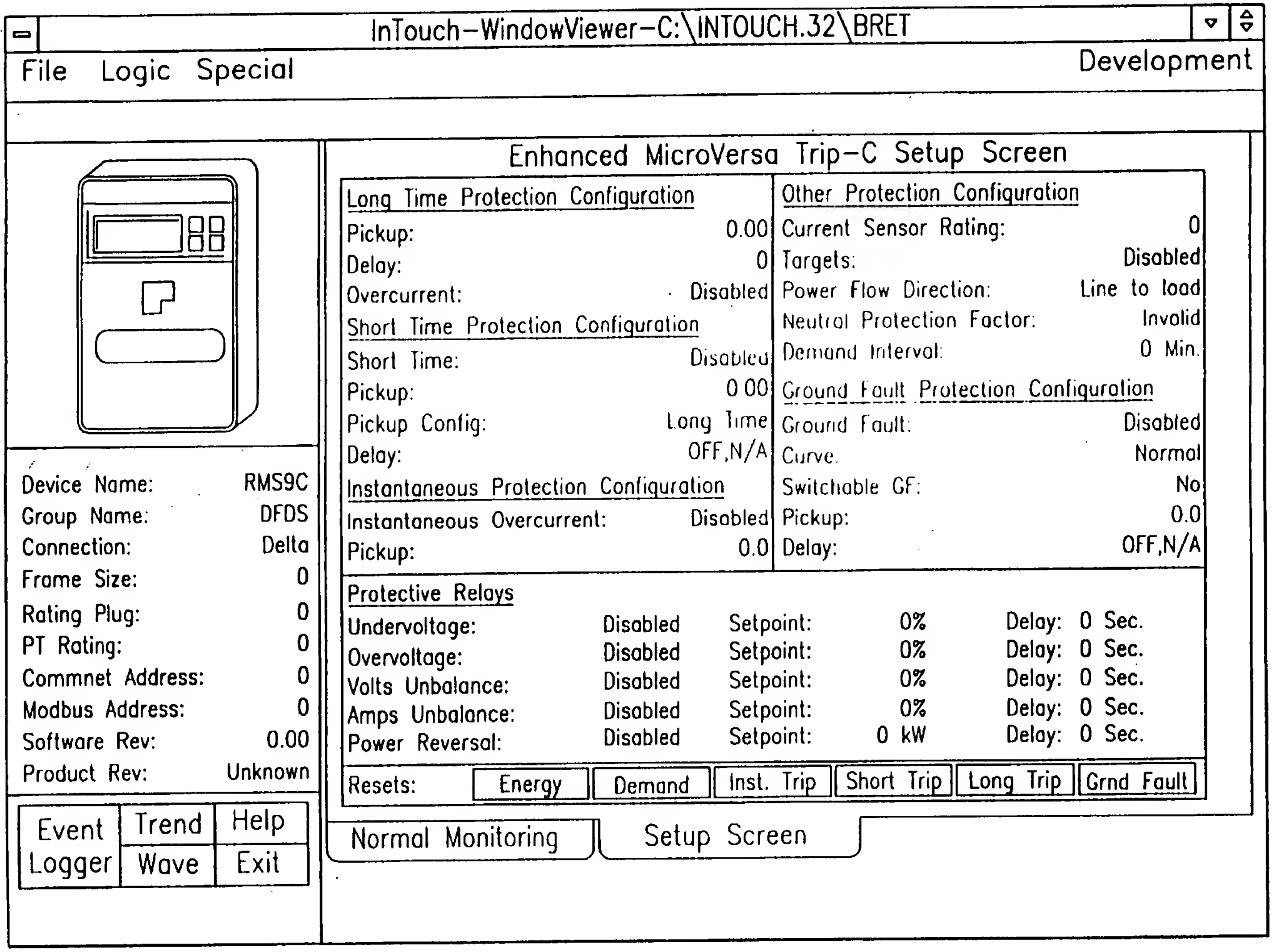
	A	B	C	Total
Amps:	0.00	0.00	0.00	
Volts L–N:	0.00	0.00	0.00	
kW:	0.00	0.00	0.00	0.00
kVAR:	0.00	0.00	0.00	0.00
kVA:	0.00	0.00	0.00	0.00

kWh:	0.00
kW Demand:	0.00
Peak kW Demand:	0.00
PF:	0.00
Frequency:	0.0
Breaker Status:	Open

Volts A–B:	0.00
Volts B–C:	0.00
Volts C–A:	0.00

Device Name:	RMS9B
Group Name:	BSD
Connection:	Delta
Frame Size:	0
Rating Plug:	0
PT Rating:	0
Commnet Address:	0
Modbus Address:	0
Software Rev:	0.00
Product Rev:	0.00

FIG. 22



Enhanced MicroVersa Trip–C Setup Screen

Device Name:	RMS9C
Group Name:	DFDS
Connection:	Delta
Frame Size:	0
Rating Plug:	0
PT Rating:	0
Commnet Address:	0
Modbus Address:	0
Software Rev:	0.00
Product Rev:	Unknown

Long Time Protection Configuration

Pickup:	0.00
Delay:	0
Overcurrent:	Disabled

Short Time Protection Configuration

Short Time:	Disabled
Pickup:	0.00
Pickup Config:	Long Time
Delay:	OFF,N/A

Instantaneous Protection Configuration

Instantaneous Overcurrent:	Disabled
Pickup:	0.0

Other Protection Configuration

Current Sensor Rating:	0
Targets:	Disabled
Power Flow Direction:	Line to load
Neutral Protection Factor:	Invalid
Demand Interval:	0 Min.

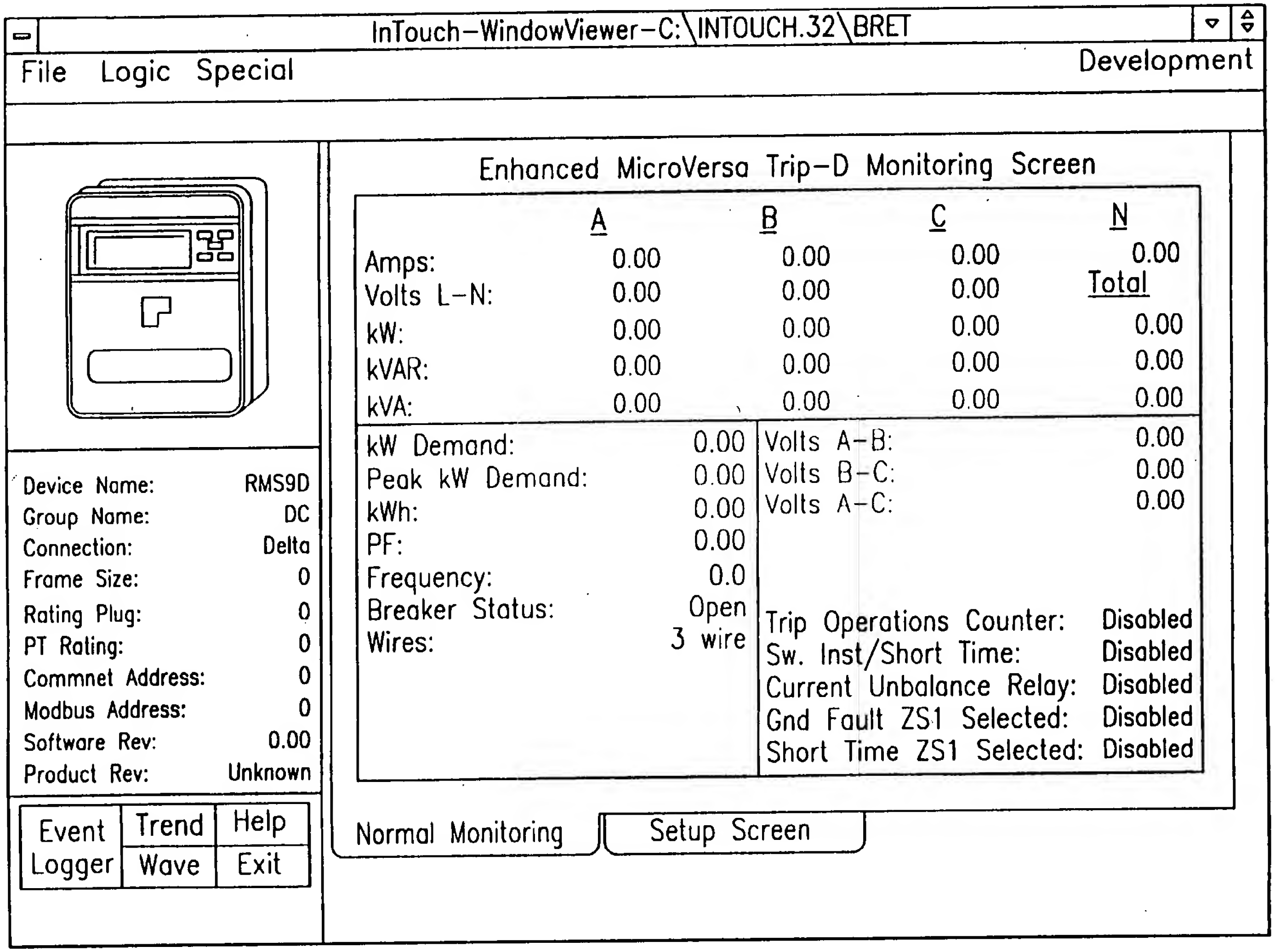
Ground Fault Protection Configuration

Ground Fault:	Disabled
Curve:	Normal
Switchable GF:	No
Pickup:	0.0
Delay:	OFF,N/A

Protective Relays

Undervoltage:	Disabled	Setpoint:	0%	Delay:	0 Sec.
Overvoltage:	Disabled	Setpoint:	0%	Delay:	0 Sec.
Volts Unbalance:	Disabled	Setpoint:	0%	Delay:	0 Sec.
Amps Unbalance:	Disabled	Setpoint:	0%	Delay:	0 Sec.
Power Reversal:	Disabled	Setpoint:	0 kW	Delay:	0 Sec.

FIG. 23



Device Name: RMS9D
Group Name: DC
Connection: Delta
Frame Size: 0
Rating Plug: 0
PT Rating: 0
Commnet Address: 0
Modbus Address: 0
Software Rev: 0.00
Product Rev: Unknown

Enhanced MicroVersa Trip–D Monitoring Screen

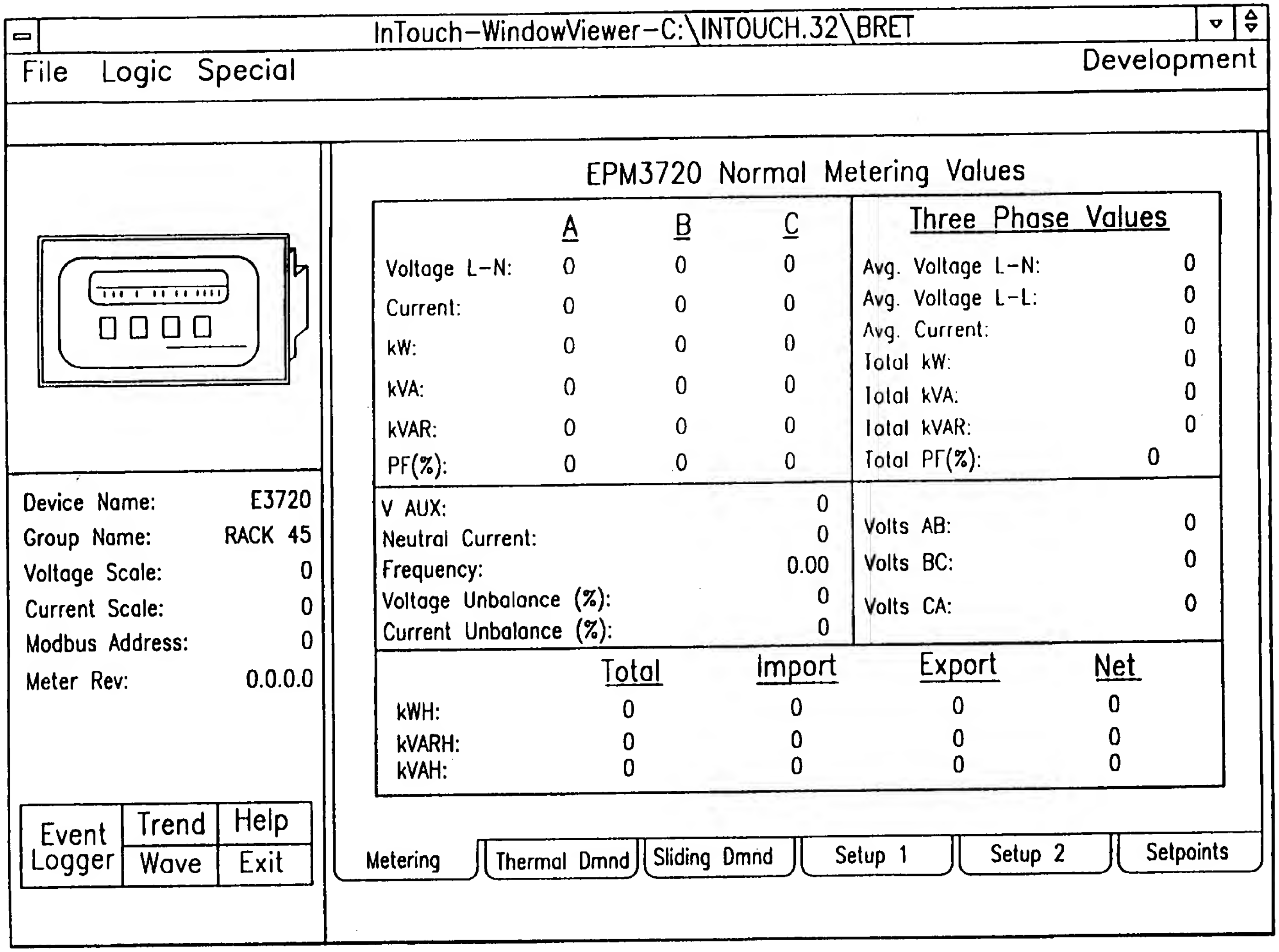
	A	B	C	N
Amps:	0.00	0.00	0.00	0.00
				Total
Volts L–N:	0.00	0.00	0.00	
kW:	0.00	0.00	0.00	0.00
kVAR:	0.00	0.00	0.00	0.00
kVA:	0.00	0.00	0.00	0.00

kW Demand: 0.00
Peak kW Demand: 0.00
kWh: 0.00
PF: 0.00
Frequency: 0.0
Breaker Status: Open
Wires: 3 wire

Volts A–B: 0.00
Volts B–C: 0.00
Volts A–C: 0.00

Trip Operations Counter: Disabled
Sw. Inst/Short Time: Disabled
Current Unbalance Relay: Disabled
Gnd Fault ZS1 Selected: Disabled
Short Time ZS1 Selected: Disabled

FIG. 24



Device Name: E3720
Group Name: RACK 45
Voltage Scale: 0
Current Scale: 0
Modbus Address: 0
Meter Rev: 0.0.0.0

EPM3720 Normal Metering Values

	A	B	C
Voltage L–N:	0	0	0
Current:	0	0	0
kW:	0	0	0
kVA:	0	0	0
kVAR:	0	0	0
PF(%):	0	0	0

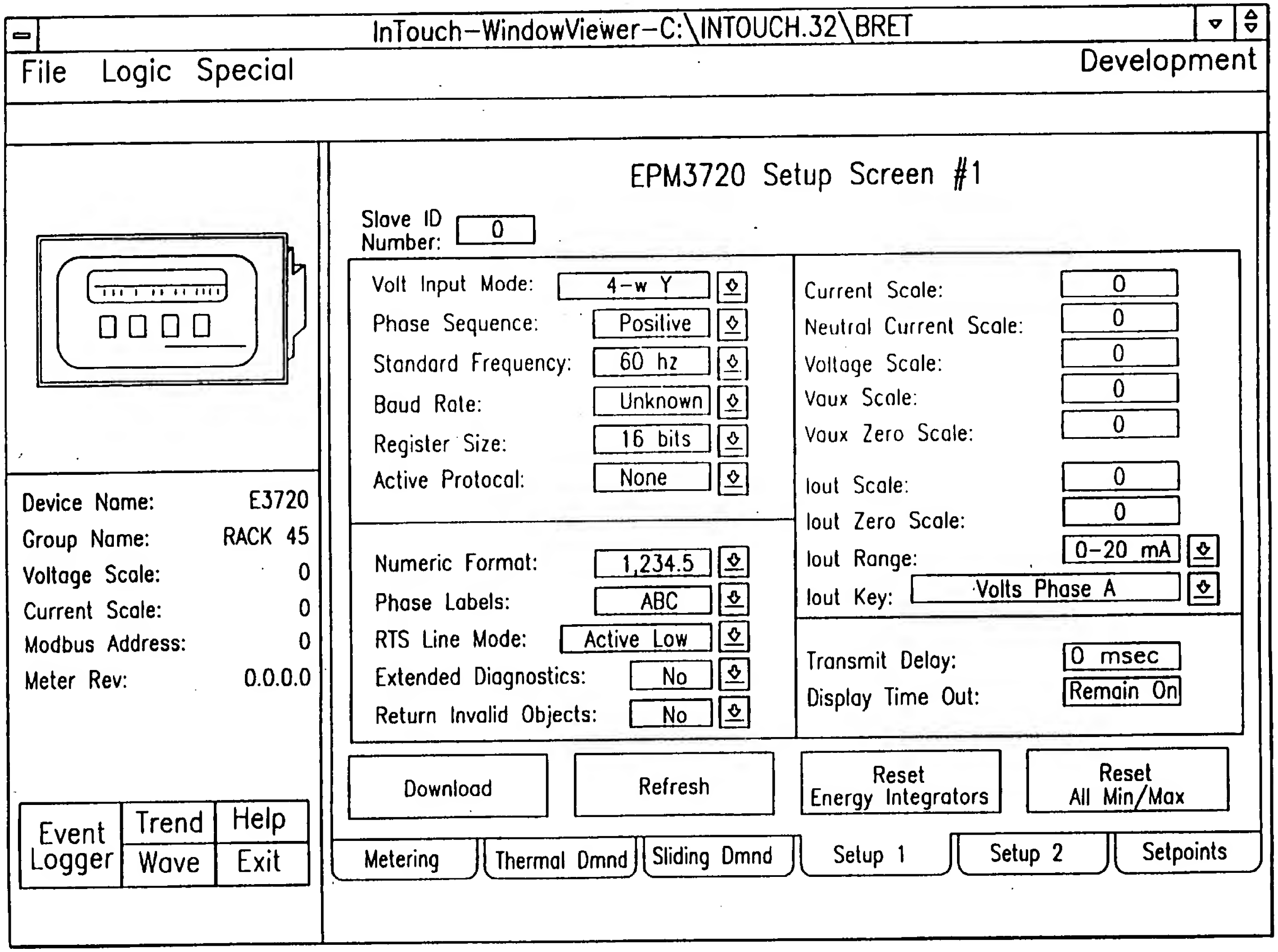
Three Phase Values	
Avg. Voltage L–N:	0
Avg. Voltage L–L:	0
Avg. Current:	0
Total kW:	0
Total kVA:	0
Total kVAR:	0
Total PF(%):	0

V AUX: 0
Neutral Current: 0
Frequency: 0.00
Voltage Unbalance (%): 0
Current Unbalance (%): 0

Volts AB: 0
Volts BC: 0
Volts CA: 0

	Total	Import	Export	Net
kWH:	0	0	0	0
kVARH:	0	0	0	0
kVAH:	0	0	0	0

FIG. 25



InTouch–WindowViewer–C:\INTOUCH.32\BRET

EPM3720 Setup Screen #1

Slave ID Number: 0

Volt Input Mode: 4–w Y
Phase Sequence: Positive
Standard Frequency: 60 hz
Baud Rate: Unknown
Register Size: 16 bits
Active Protocol: None

Numeric Format: 1,234.5
Phase Labels: ABC
RTS Line Mode: Active Low
Extended Diagnostics: No
Return Invalid Objects: No

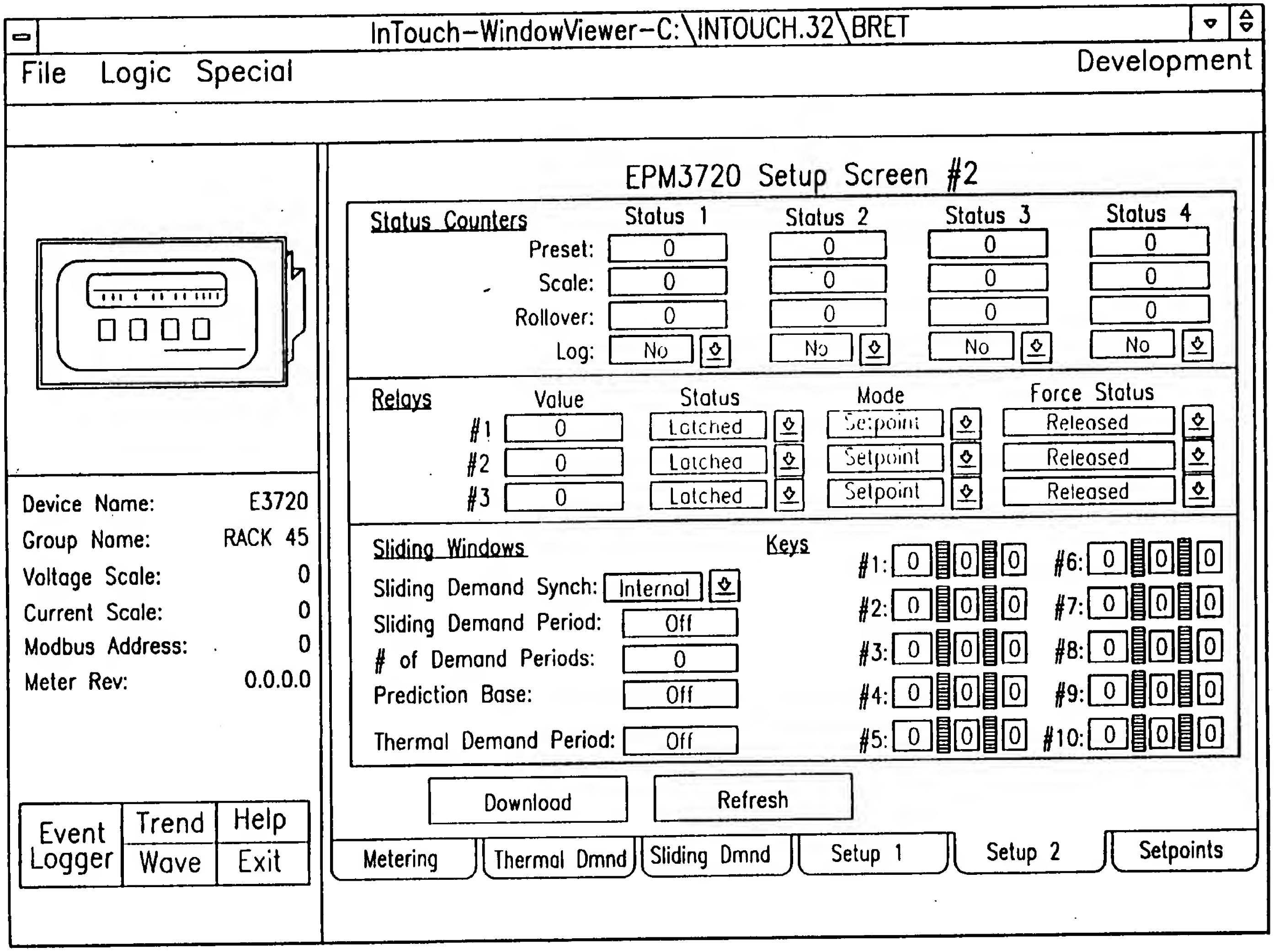
Current Scale: 0
Neutral Current Scale: 0
Voltage Scale: 0
Vaux Scale: 0
Vaux Zero Scale: 0

Iout Scale: 0
Iout Zero Scale: 0
Iout Range: 0–20 mA
Iout Key: Volts Phase A

Transmit Delay: 0 msec
Display Time Out: Remain On

Device Name: E3720
Group Name: RACK 45
Voltage Scale: 0
Current Scale: 0
Modbus Address: 0
Meter Rev: 0.0.0.0

FIG. 26



EPM3720 Setup Screen #2

Device Name: E3720
Group Name: RACK 45
Voltage Scale: 0
Current Scale: 0
Modbus Address: 0
Meter Rev: 0.0.0.0

Status Counters

	Status 1	Status 2	Status 3	Status 4
Preset:	0	0	0	0
Scale:	0	0	0	0
Rollover:	0	0	0	0
Log:	No	No	No	No

Relays

	Value	Status	Mode	Force Status
#1	0	Latched	Setpoint	Released
#2	0	Latched	Setpoint	Released
#3	0	Latched	Setpoint	Released

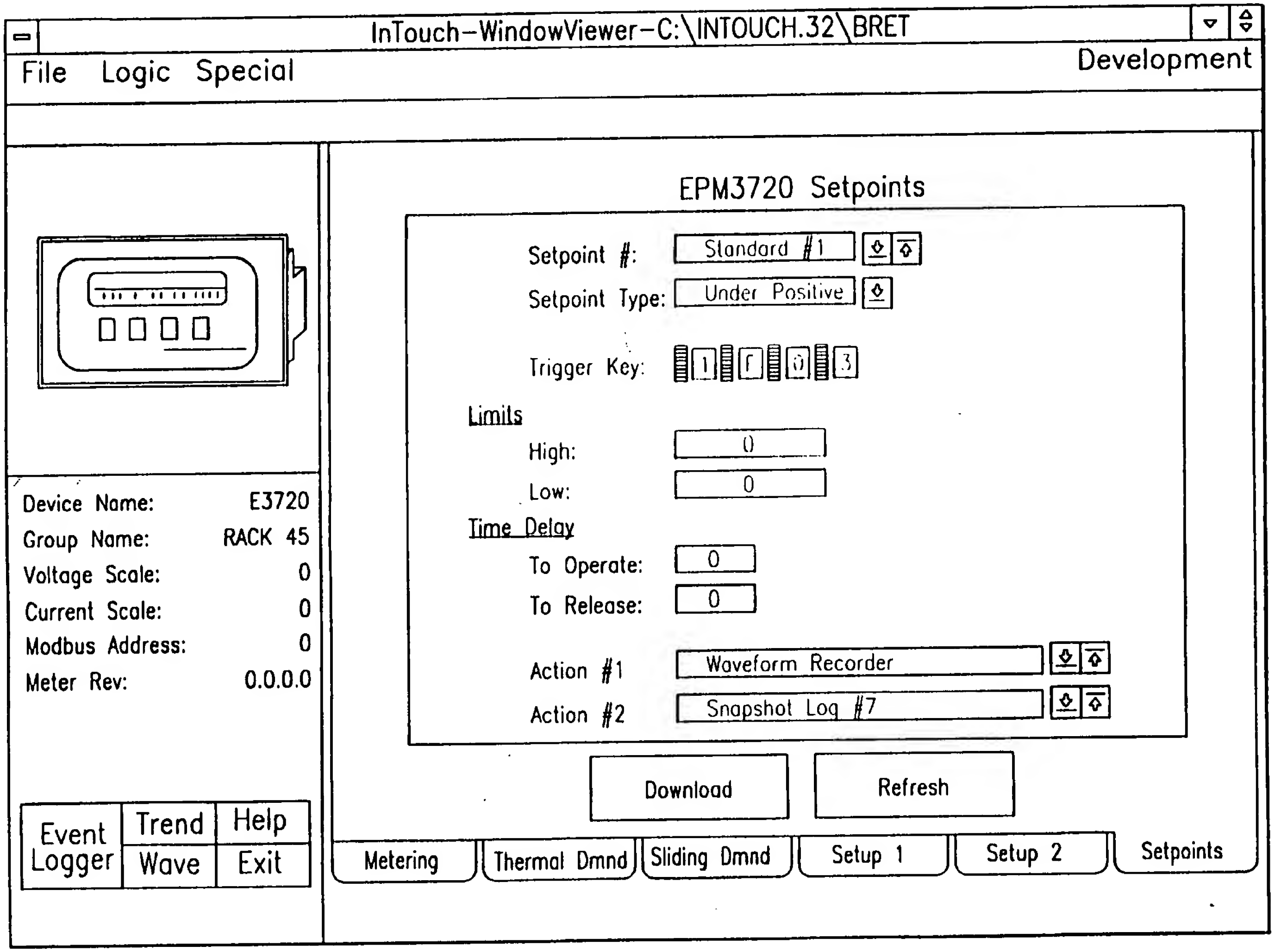
Sliding Windows

Sliding Demand Synch: Internal
Sliding Demand Period: Off
# of Demand Periods: 0
Prediction Base: Off
Thermal Demand Period: Off

Keys

#1:	0 0 0		#6:	0 0 0
#2:	0 0 0		#7:	0 0 0
#3:	0 0 0		#8:	0 0 0
#4:	0 0 0		#9:	0 0 0
#5:	0 0 0		#10:	0 0 0

FIG. 27



Device Name: E3720
Group Name: RACK 45
Voltage Scale: 0
Current Scale: 0
Modbus Address: 0
Meter Rev: 0.0.0.0

EPM3720 Setpoints

Setpoint #: Standard #1
Setpoint Type: Under Positive
Trigger Key: 1 [0 3

Limits
High: 0
Low: 0

Time Delay
To Operate: 0
To Release: 0

Action #1 Waveform Recorder
Action #2 Snapshot Log #7

FIG. 28

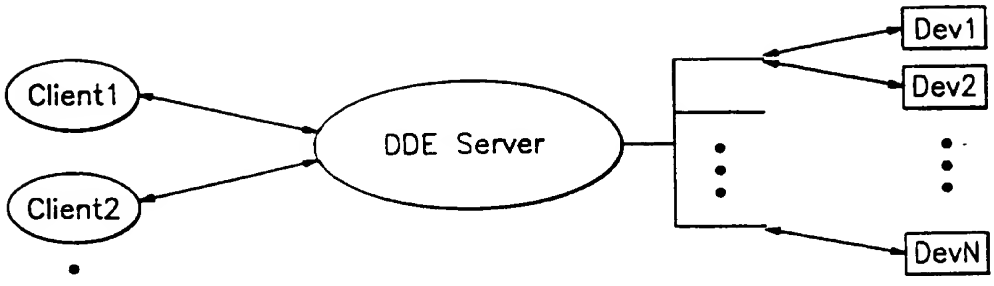
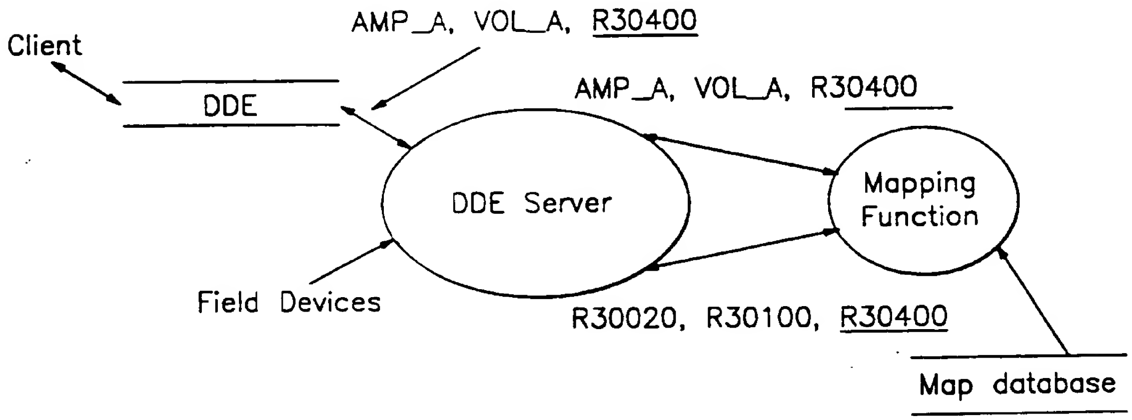


FIG. 29



Register Mapping Scheme

AMP_A => Current of phase A for a meter identified by DDE topic,
Register address R30020

VOL_A => Voltage of phase A for a meter identified by DDE topic,
Register address R30100

R30400 => An item addressed directly with register address.
No conversion required.

FIG. 30

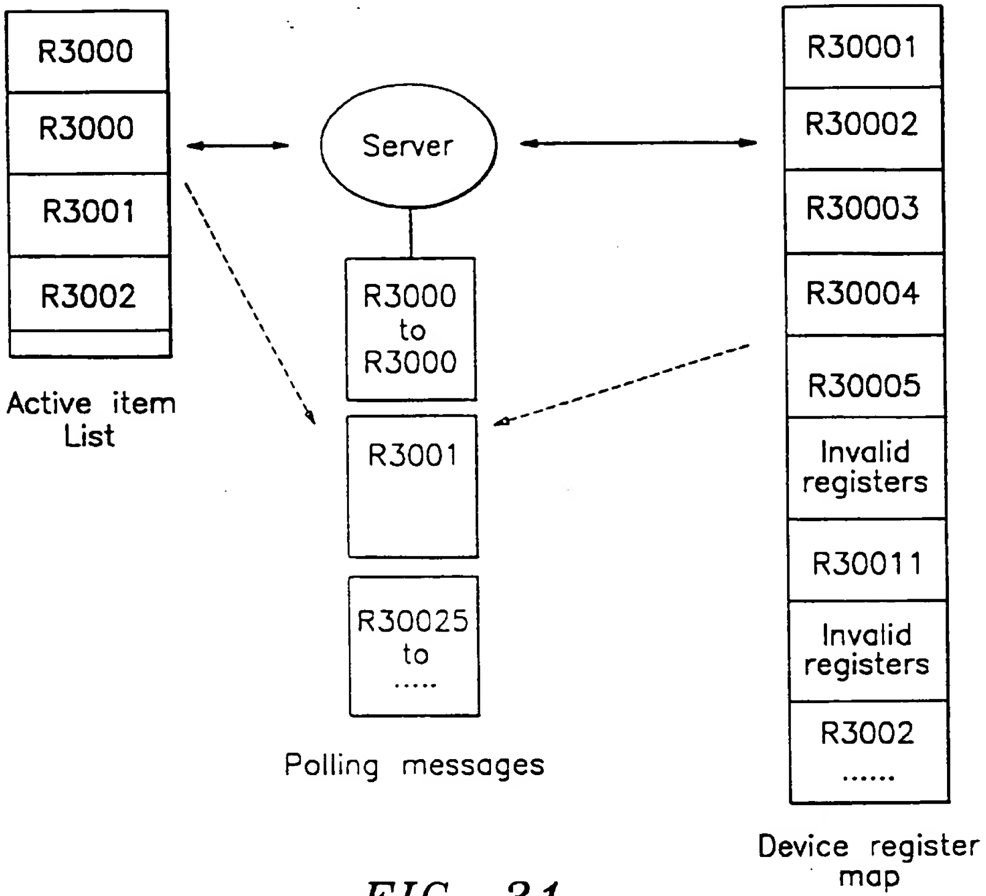


FIG. 31

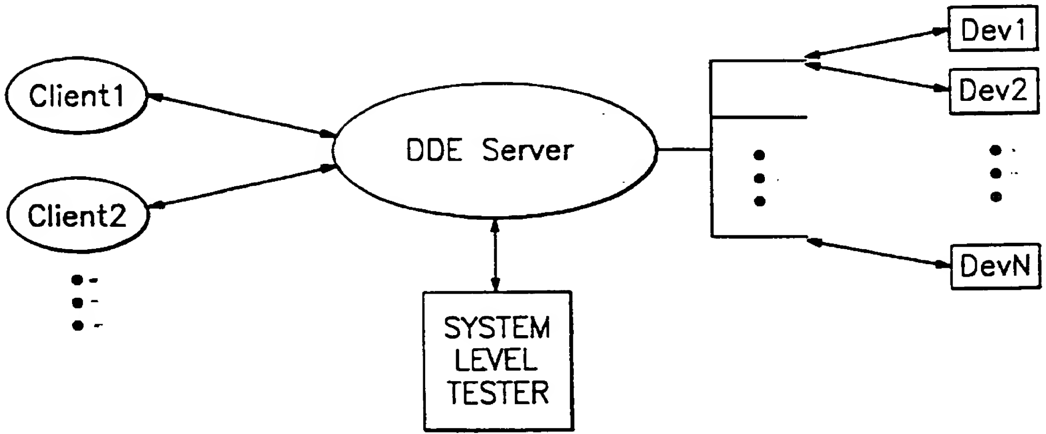


FIG. 31A

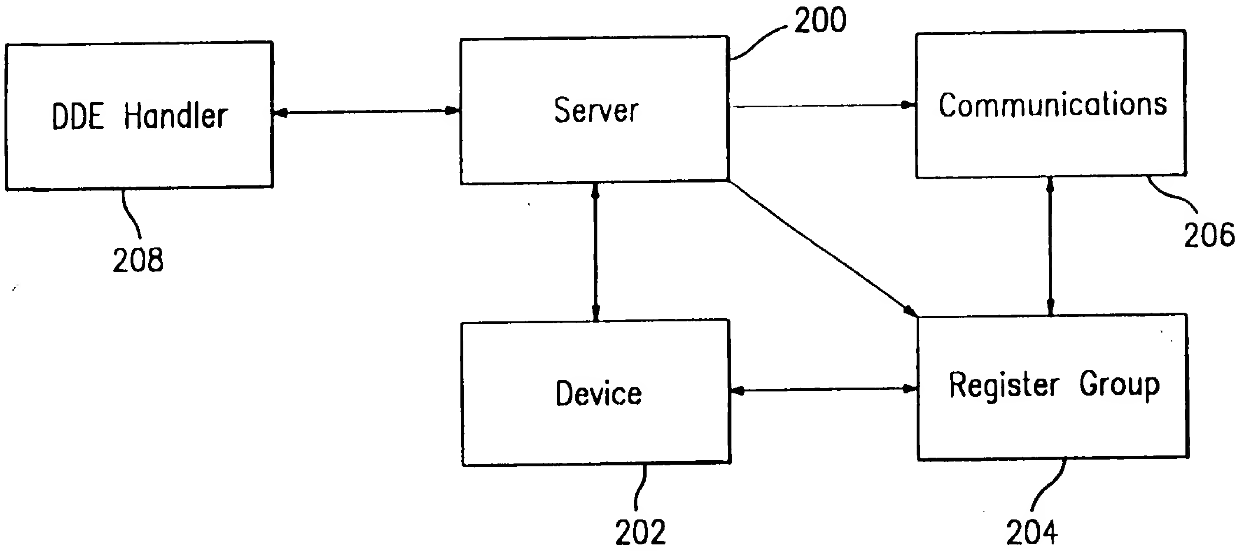

DDE Handler
208
Server
200
Communications
206
Device
202
Register Group
204

FIG. 32

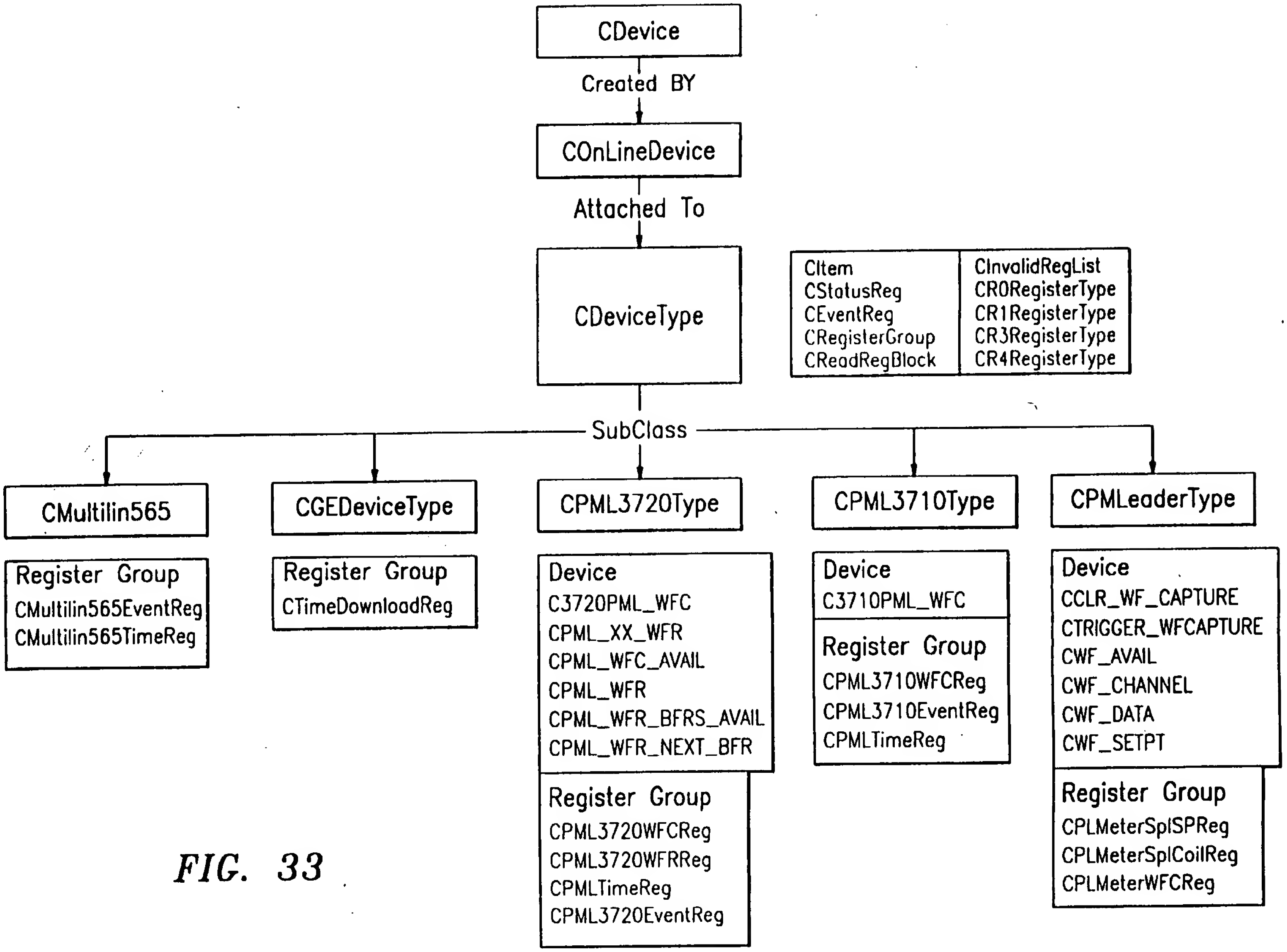

CDevice
Created BY
COnLineDevice
Attached To
CDeviceType
CItem
CStatusReg
CEventReg
CRegisterGroup
CReadRegBlock
CInvalidRegList
CR0RegisterType
CR1RegisterType
CR3RegisterType
CR4RegisterType
SubClass
CMultilin565
Register Group
CMultilin565EventReg
CMultilin565TimeReg
CGEDeviceType
Register Group
CTimeDownloadReg
CPML3720Type
Device
C3720PML_WFC
CPML_XX_WFR
CPML_WFC_AVAIL
CPML_WFR
CPML_WFR_BFRS_AVAIL
CPML_WFR_NEXT_BFR
Register Group
CPML3720WFCReg
CPML3720WFRReg
CPMLTimeReg
CPML3720EventReg
CPML3710Type
Device
C3710PML_WFC
Register Group
CPML3710WFCReg
CPML3710EventReg
CPMLTimeReg
CPMLeaderType
Device
CCLR_WF_CAPTURE
CTRIGGER_WFCAPTURE
CWF_AVAIL
CWF_CHANNEL
CWF_DATA
CWF_SETPT
Register Group
CPLMeterSplSPReg
CPLMeterSplCoilReg
CPLMeterWFCReg

FIG. 33

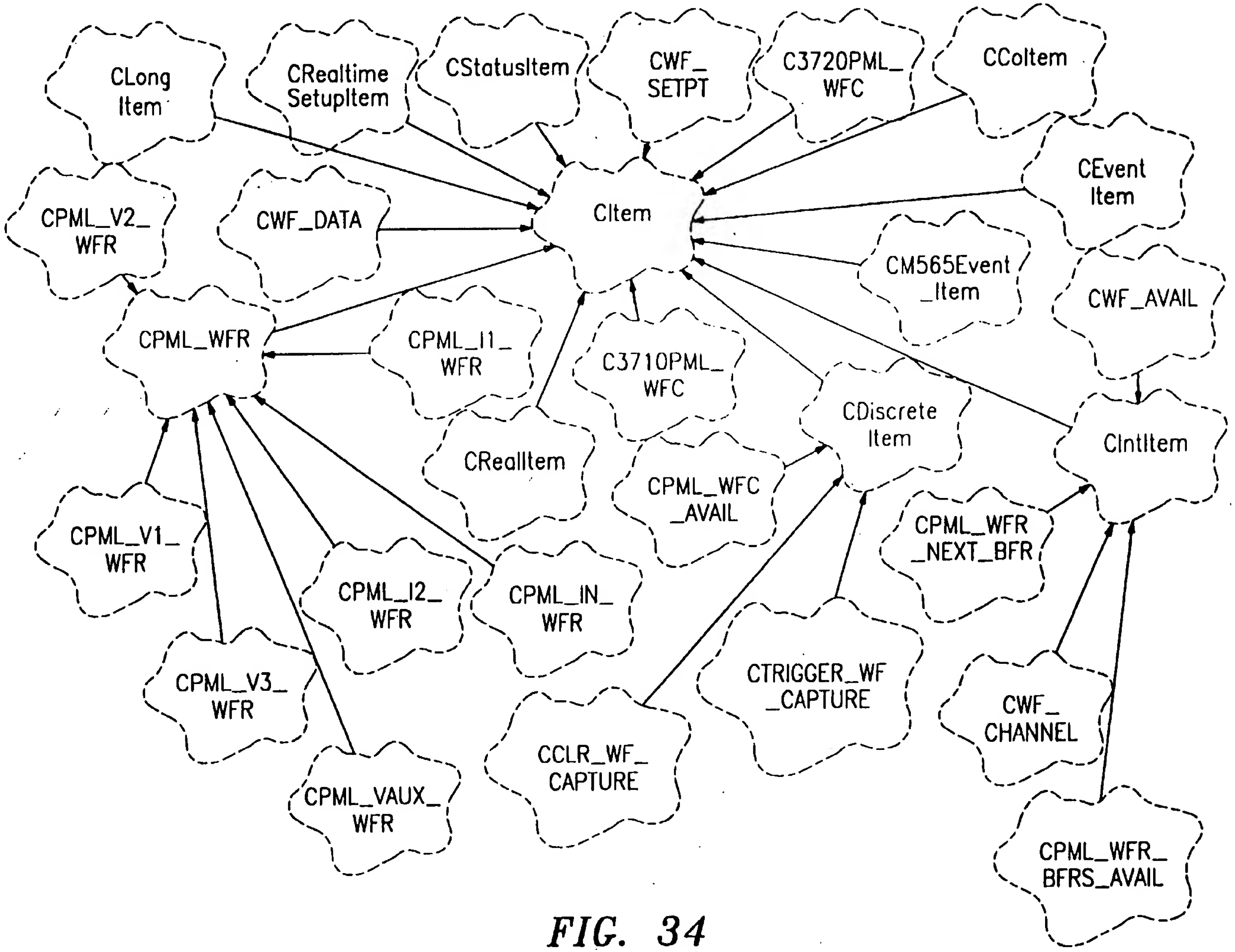

CLong Item
CRealtime SetupItem
CStatusItem
CWF_ SETPT
C3720PML_ WFC
CCoItem
CEvent Item
CPML_V2_ WFR
CWF_DATA
CItem
CM565Event _Item
CWF_AVAIL
CPML_WFR
CPML_I1_ WFR
C3710PML_ WFC
CDiscrete Item
CIntItem
CRealItem
CPML_WFC _AVAIL
CPML_WFR _NEXT_BFR
CPML_V1_ WFR
CPML_I2_ WFR
CPML_IN_ WFR
CPML_V3_ WFR
CTRIGGER_WF _CAPTURE
CWF_ CHANNEL
CCLR_WF_ CAPTURE
CPML_VAUX_ WFR
CPML_WFR_ BFRS_AVAIL

FIG. 34

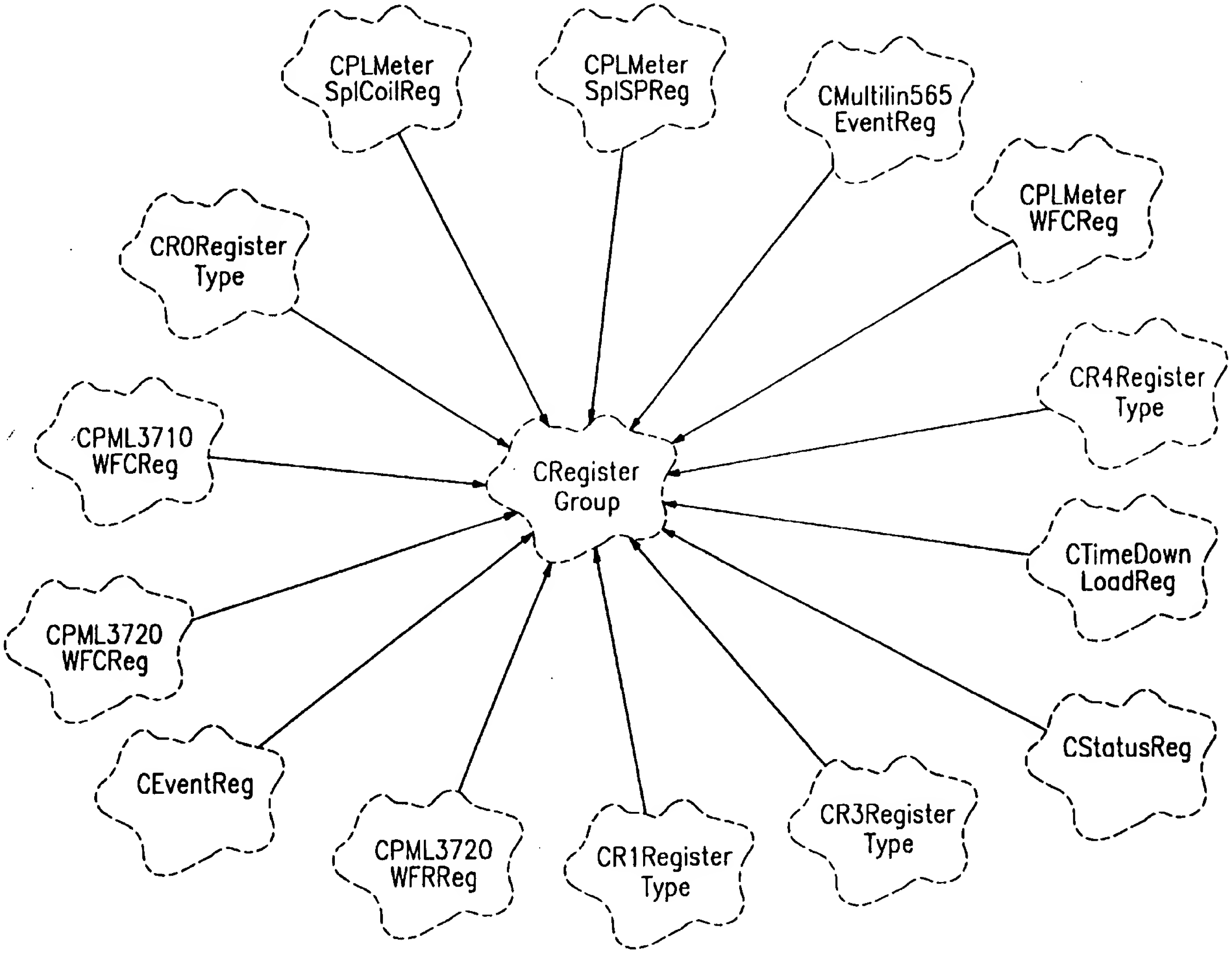

CPLMeter SplCoilReg
CPLMeter SplSPReg
CMultilin565 EventReg
CPLMeter WFCReg
CR0Register Type
CR4Register Type
CPML3710 WFCReg
CRegister Group
CTimeDown LoadReg
CPML3720 WFCReg
CStatusReg
CEventReg
CPML3720 WFRReg
CR1Register Type
CR3Register Type

FIG. 35

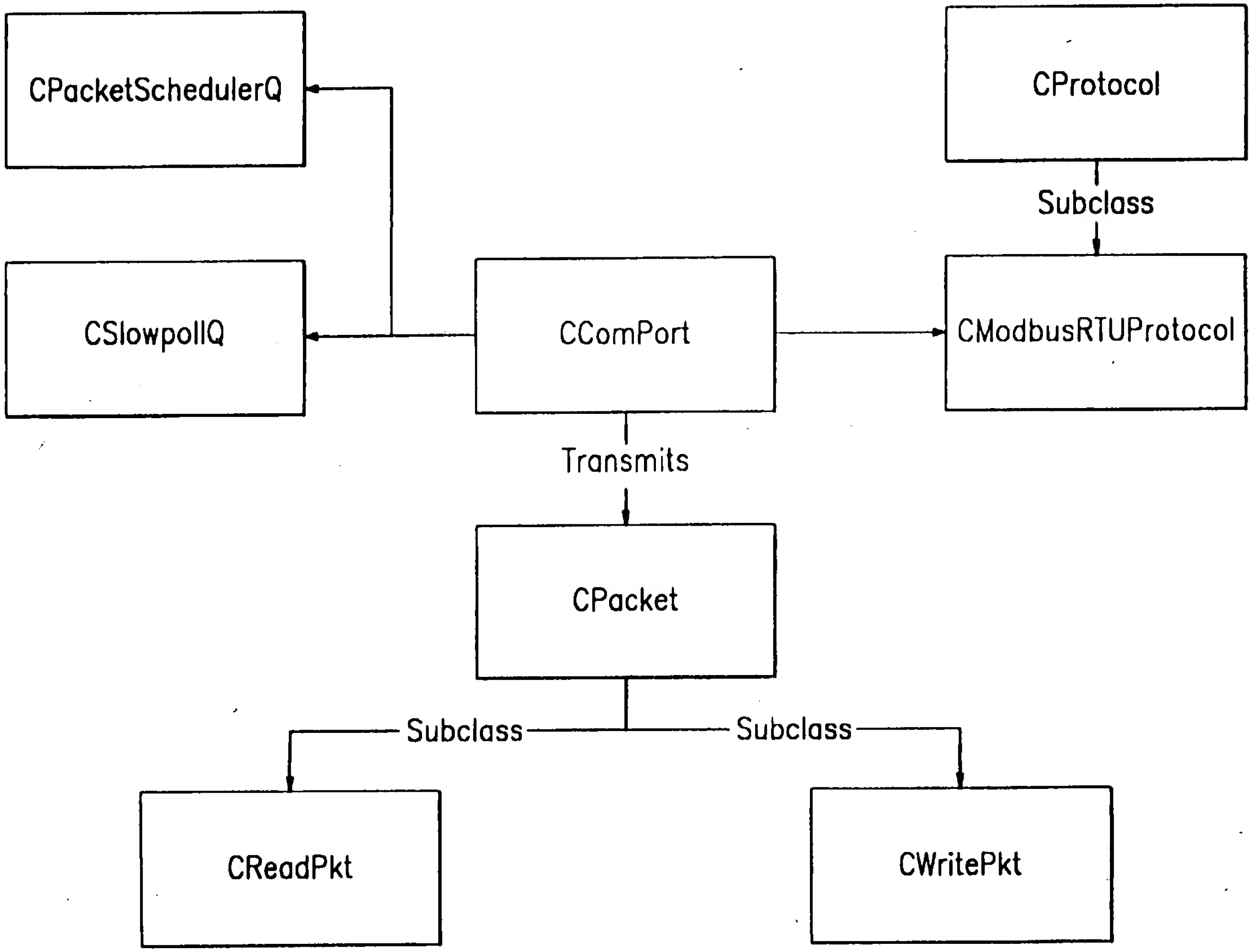

CPacketSchedulerQ
CSlowpollQ
CComPort
CProtocol
Subclass
CModbusRTUProtocol
Transmits
CPacket
Subclass
Subclass
CReadPkt
CWritePkt

FIG. 36

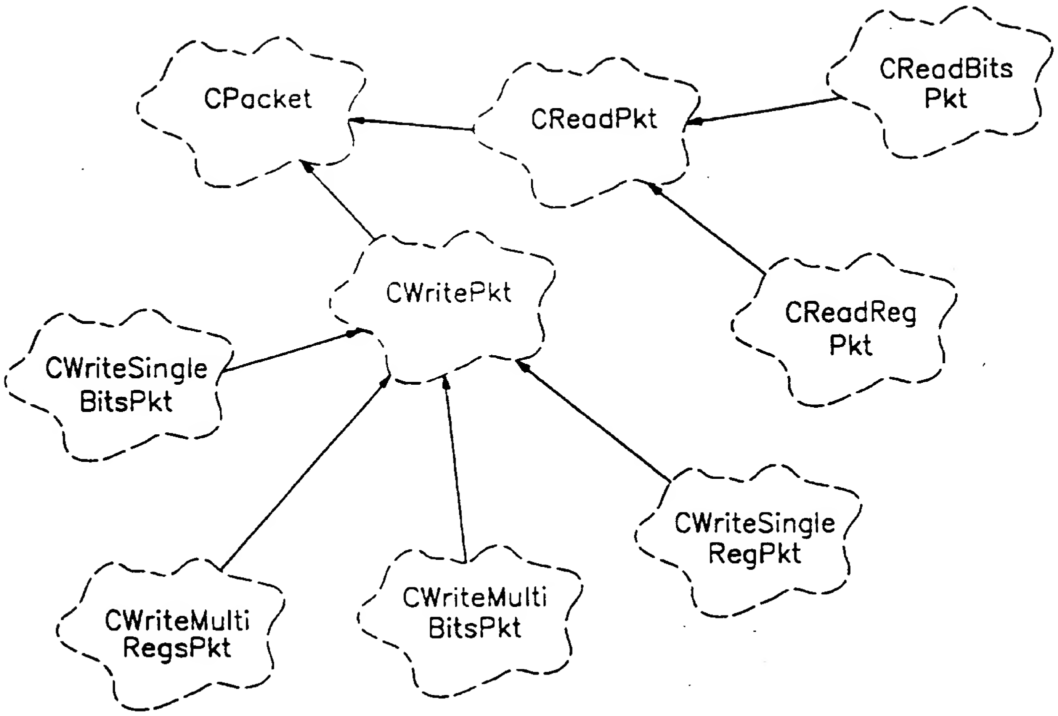

CPacket
CReadPkt
CReadBits Pkt
CWritePkt
CReadReg Pkt
CWriteSingle BitsPkt
CWriteSingle RegPkt
CWriteMulti RegsPkt
CWriteMulti BitsPkt

FIG. 37

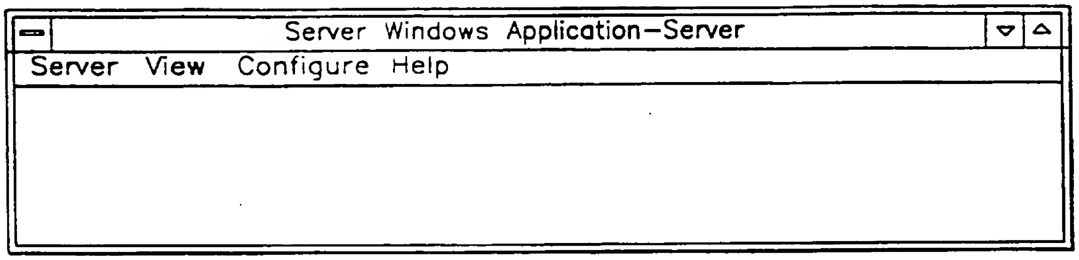

Server Windows Application–Server
Server
View
Configure
Help

FIG. 38

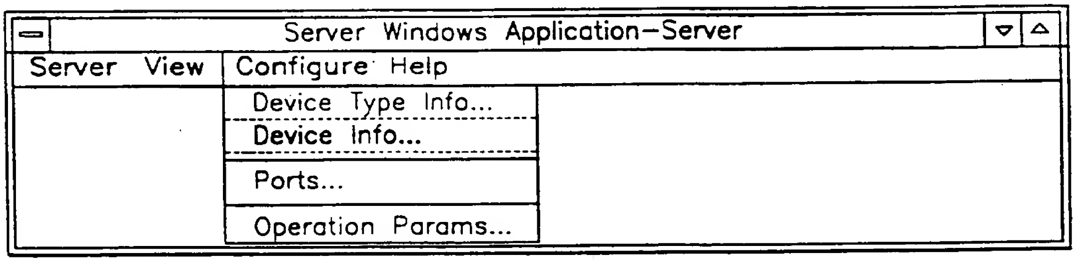
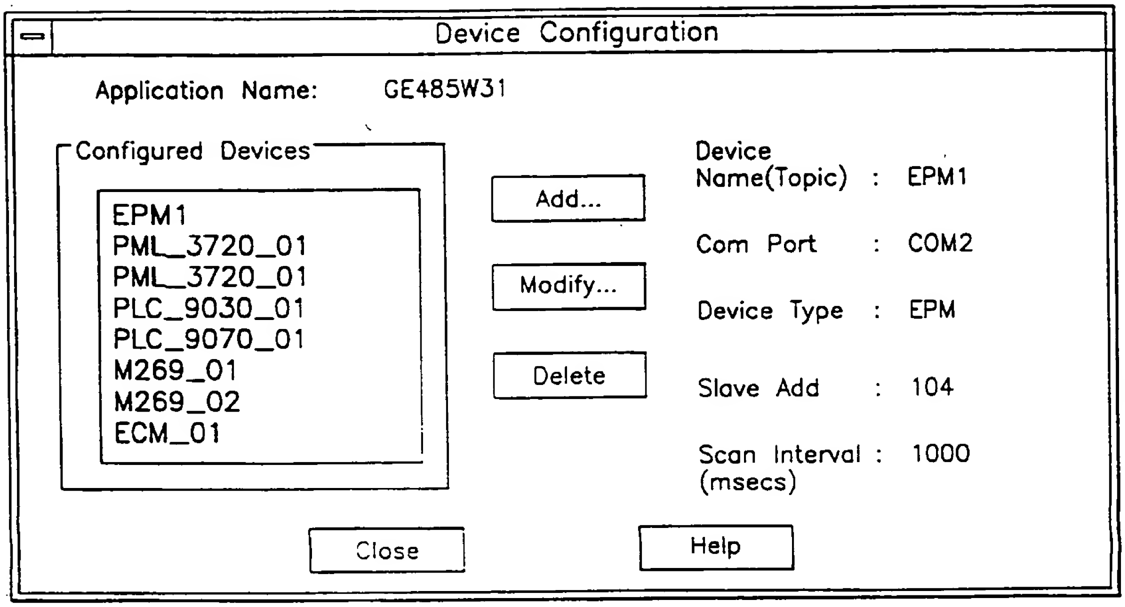

Server Windows Application–Server
Server
View
Configure
Help
Device Type Info...
Device Info...
Ports...
Operation Params...

FIG. 39



Application Name: GE485W31

EPM1
PML_3720_01
PML_3720_01
PLC_9030_01
PLC_9070_01
M269_01
M269_02
ECM_01

Device Name(Topic) : EPM1
Com Port : COM2
Device Type : EPM
Slave Add : 104
Scan Interval (msecs) : 1000

FIG. 40

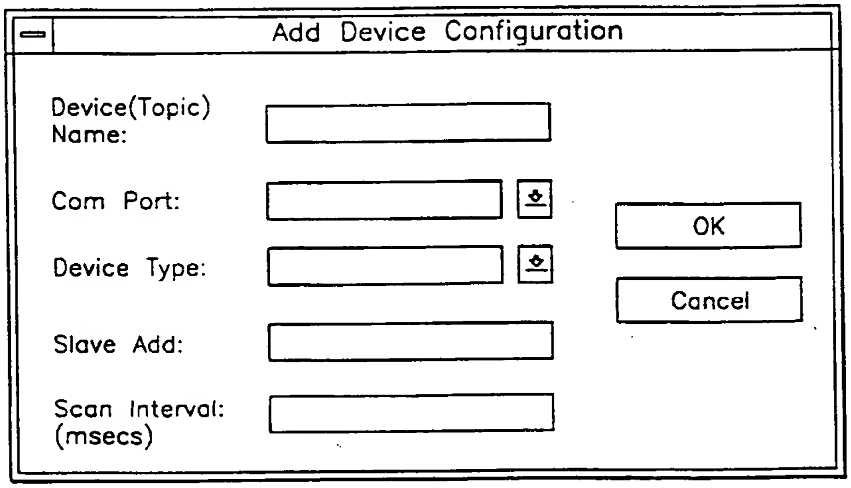
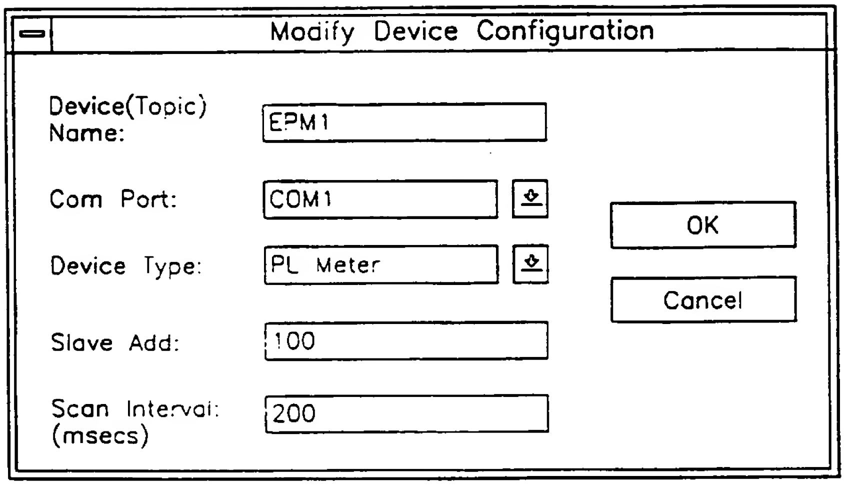


FIG. 41



Device(Topic) Name: EPM1

Com Port: COM1

Device Type: PL Meter

Slave Add: 100

Scan Interval: (msecs) 200

FIG. 42

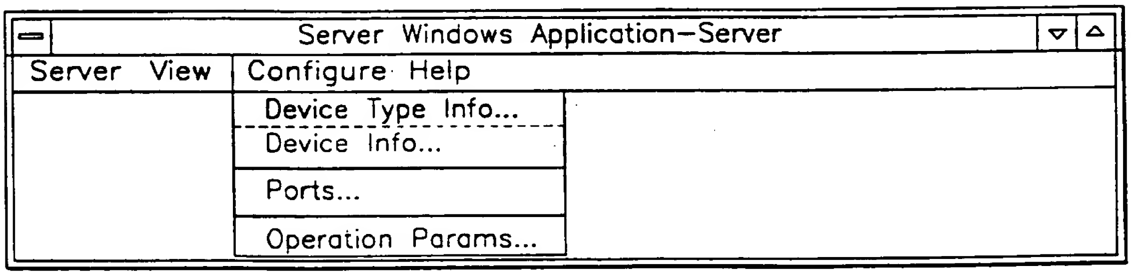
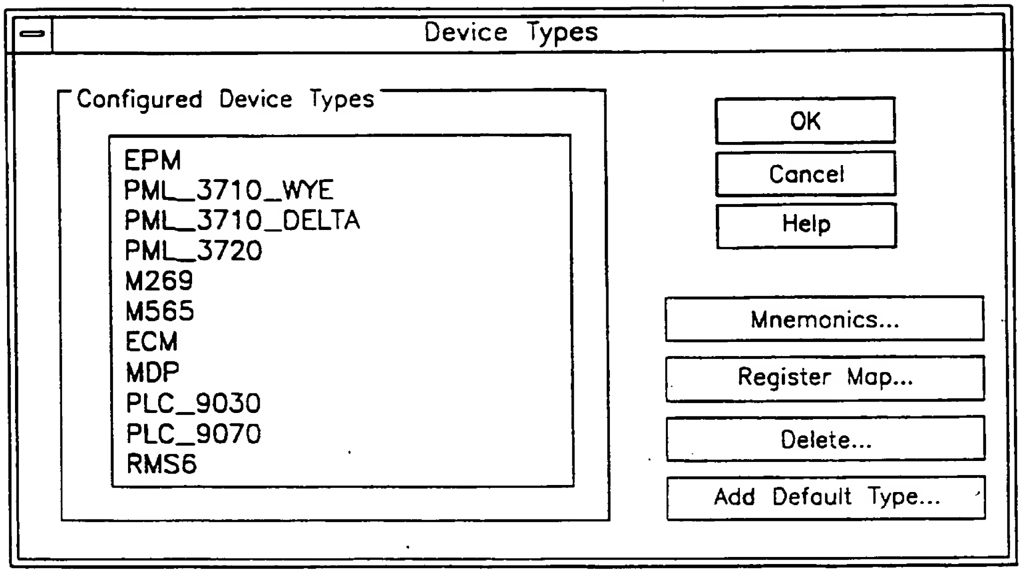
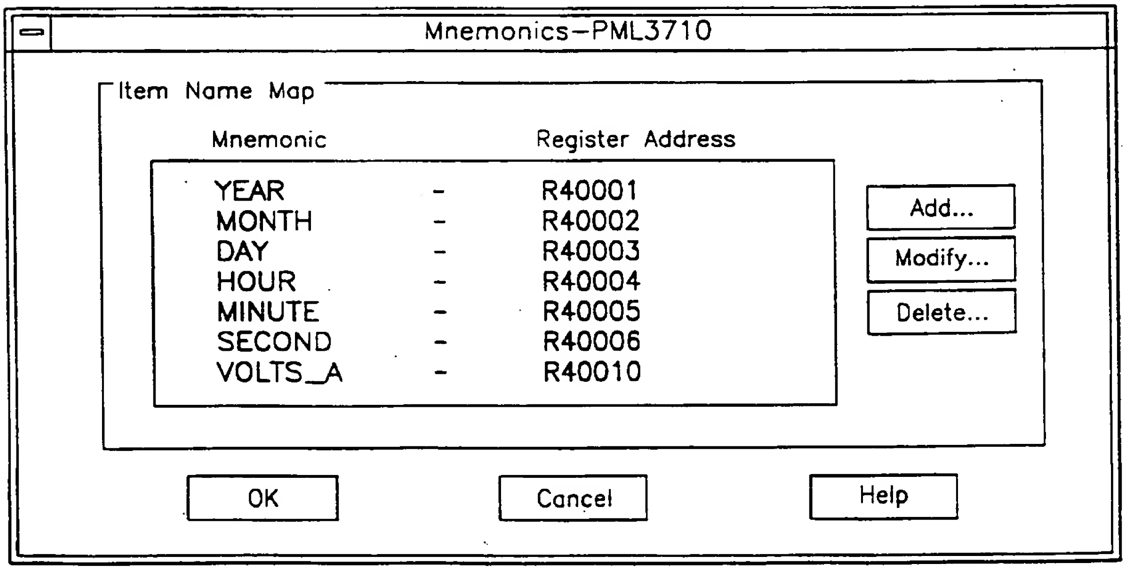


FIG. 43



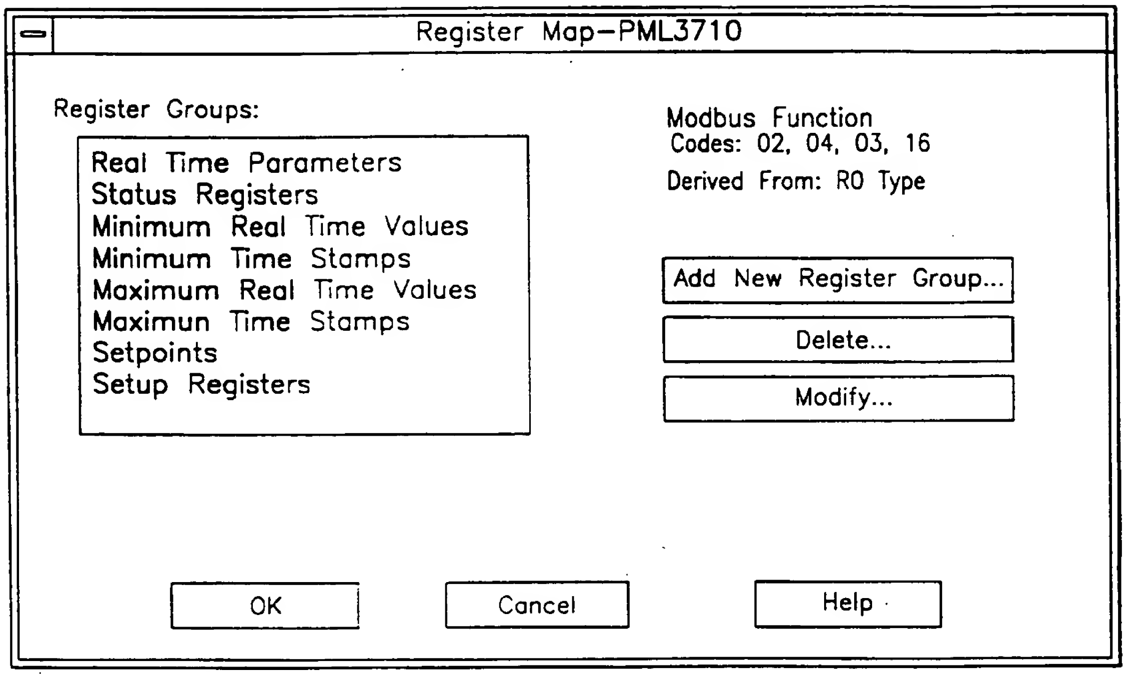
EPM
PML_3710_WYE
PML_3710_DELTA
PML_3720
M269
M565
ECM
MDP
PLC_9030
PLC_9070
RMS6

FIG. 44



Mnemonic		Register Address
YEAR	-	R40001
MONTH	-	R40002
DAY	-	R40003
HOUR	-	R40004
MINUTE	-	R40005
SECOND	-	R40006
VOLTS_A	-	R40010

FIG. 45

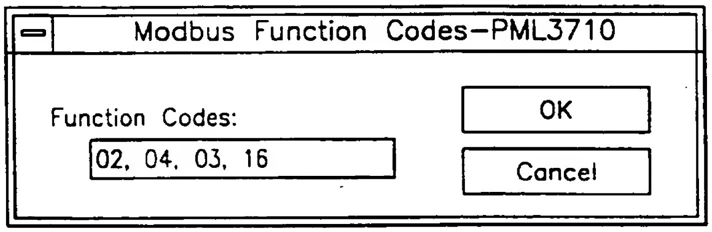


Register Groups:

Real Time Parameters
Status Registers
Minimum Real Time Values
Minimum Time Stamps
Maximum Real Time Values
Maximun Time Stamps
Setpoints
Setup Registers

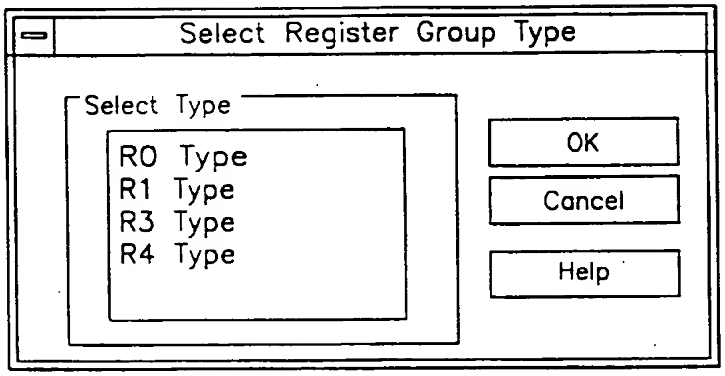
Modbus Function
Codes: 02, 04, 03, 16
Derived From: RO Type

FIG. 46



Function Codes: 02, 04, 03, 16

FIG. 47

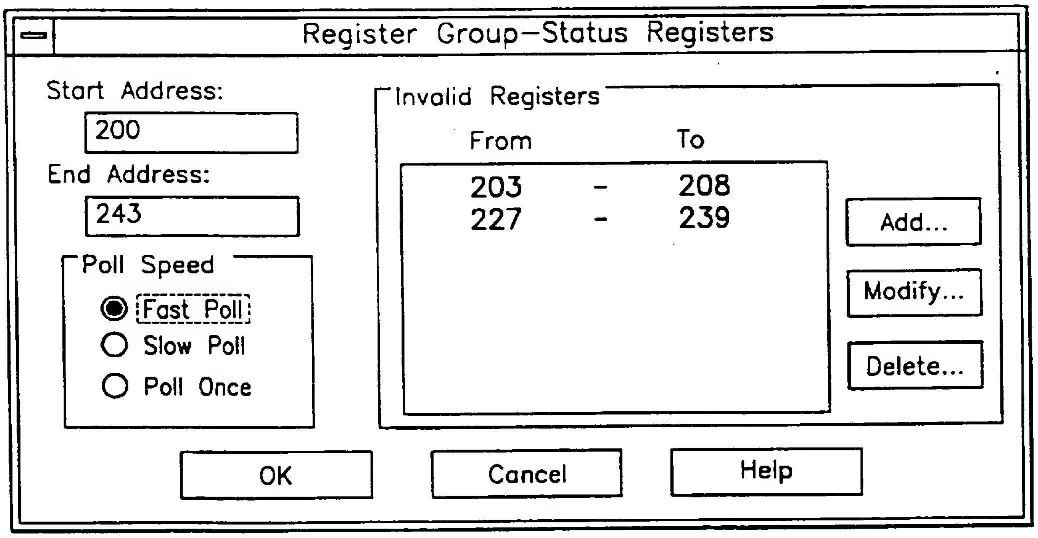


Select Register Group Type

Select Type

R0 Type
R1 Type
R3 Type
R4 Type

FIG. 48



Register Group—Status Registers

Start Address: 200

End Address: 243

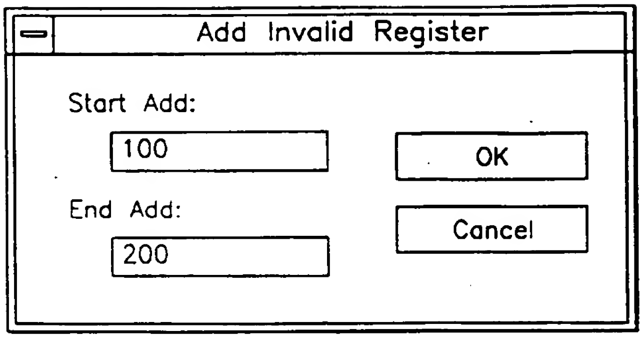
Poll Speed

● Fast Poll
○ Slow Poll
○ Poll Once

Invalid Registers

From		To
203	-	208
227	-	239

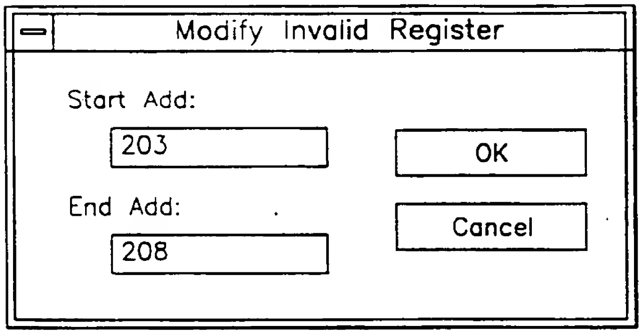
FIG. 49



Start Add: 100

End Add: 200

FIG. 50



Start Add: 203

End Add: 208

FIG. 51

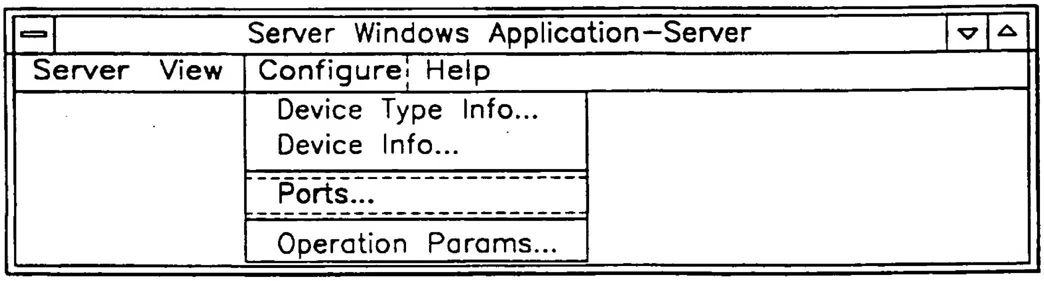
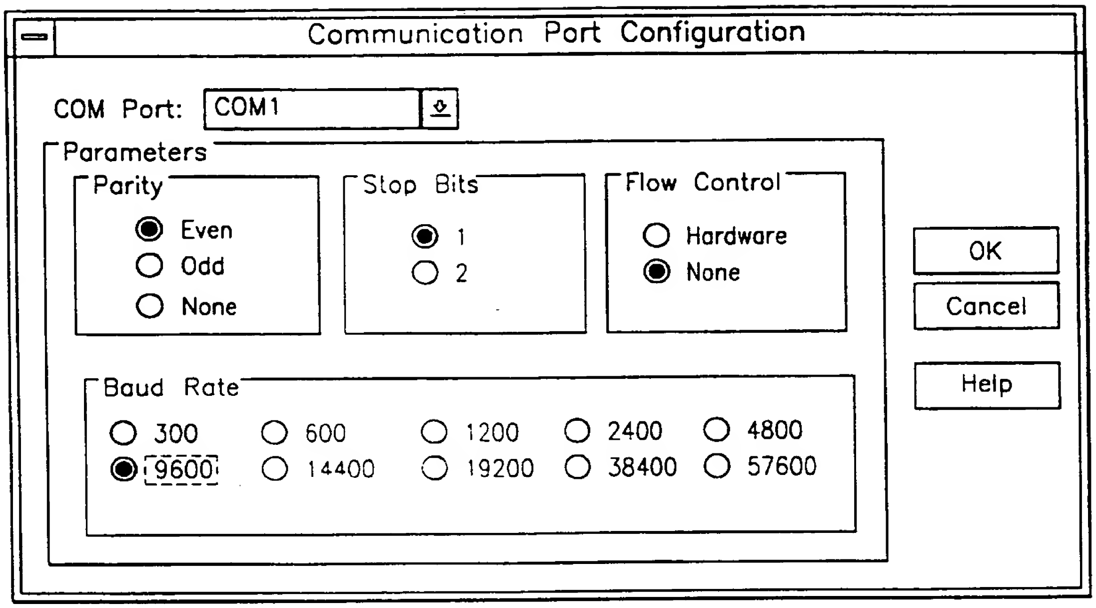


FIG. 52



COM Port: COM1

Parameters

Parity
● Even
○ Odd
○ None

Stop Bits
● 1
○ 2

Flow Control
○ Hardware
● None

Baud Rate
○ 300 ○ 600 ○ 1200 ○ 2400 ○ 4800
● 9600 ○ 14400 ○ 19200 ○ 38400 ○ 57600

FIG. 53

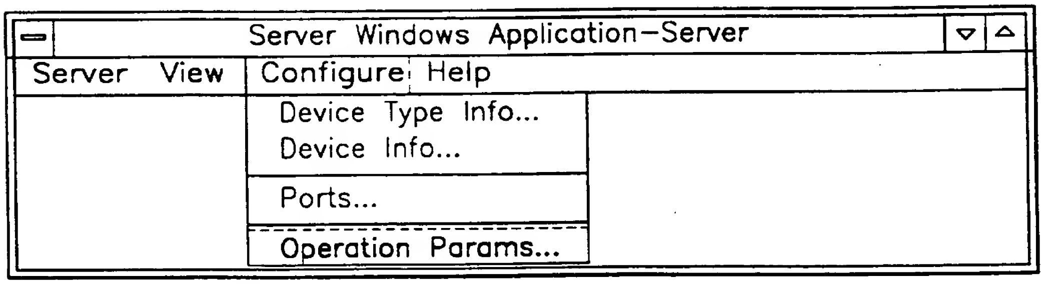
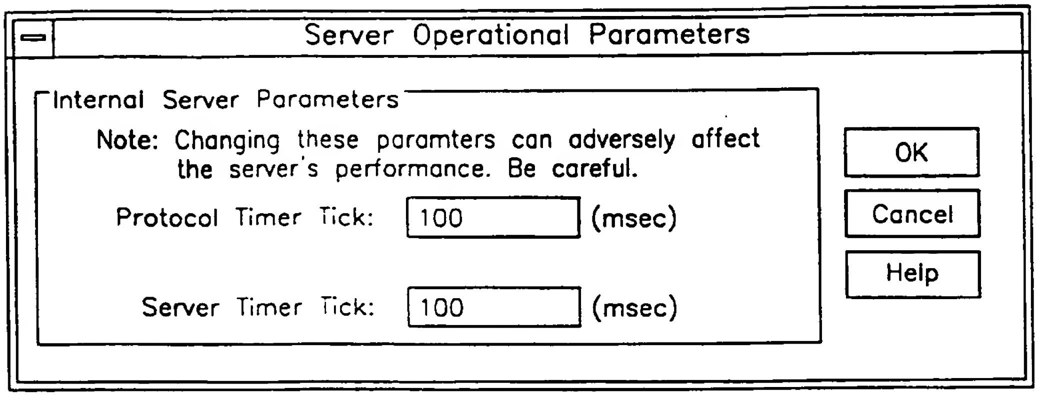


FIG. 54



Note: Changing these paramters can adversely affect the server's performance. Be careful.

Protocol Timer Tick: 100 (msec)

Server Timer Tick: 100 (msec)

FIG. 55

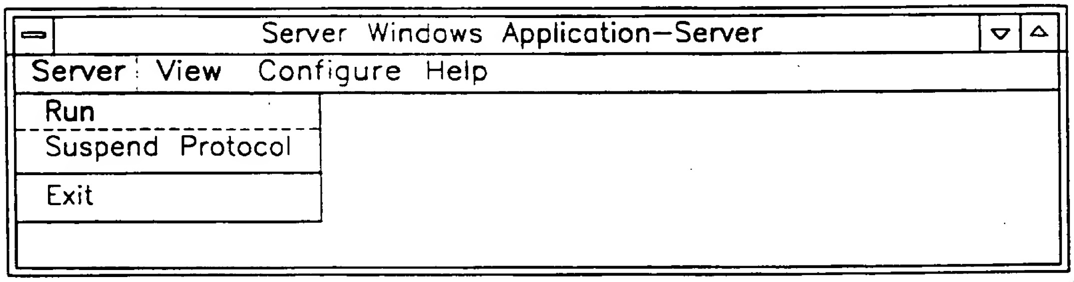


FIG. 56

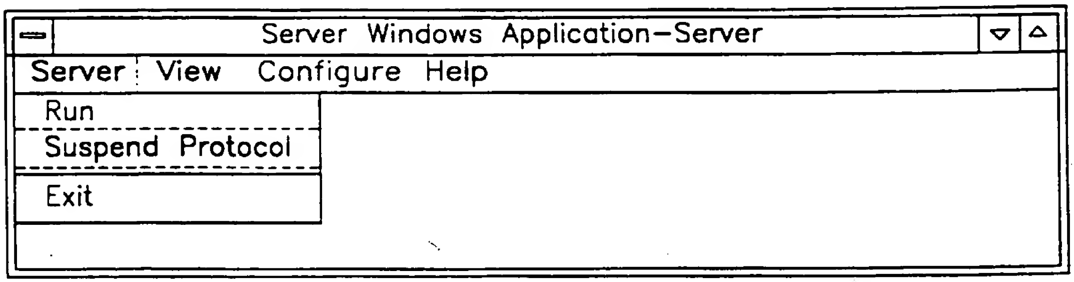


FIG. 57

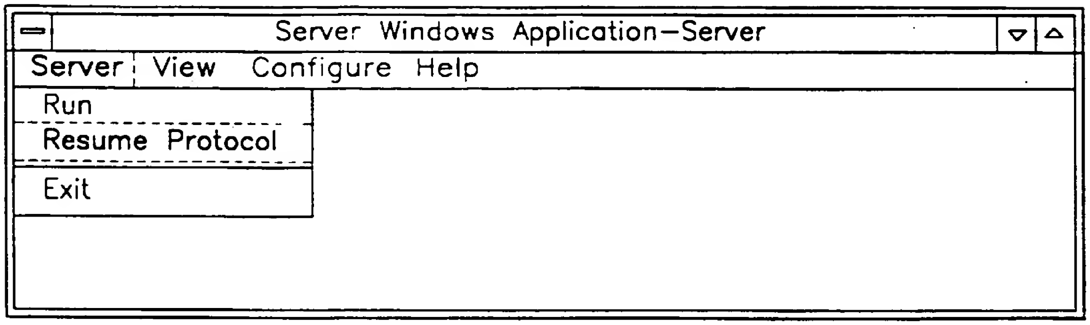

Server Windows Application–Server
Server
View
Configure
Help
Run
Resume Protocol
Exit

FIG. 58

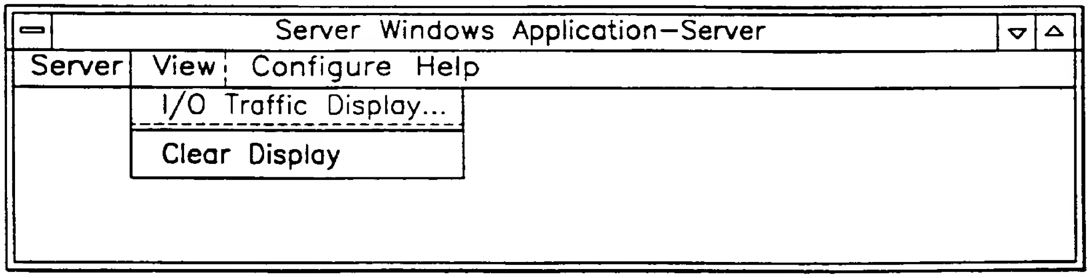
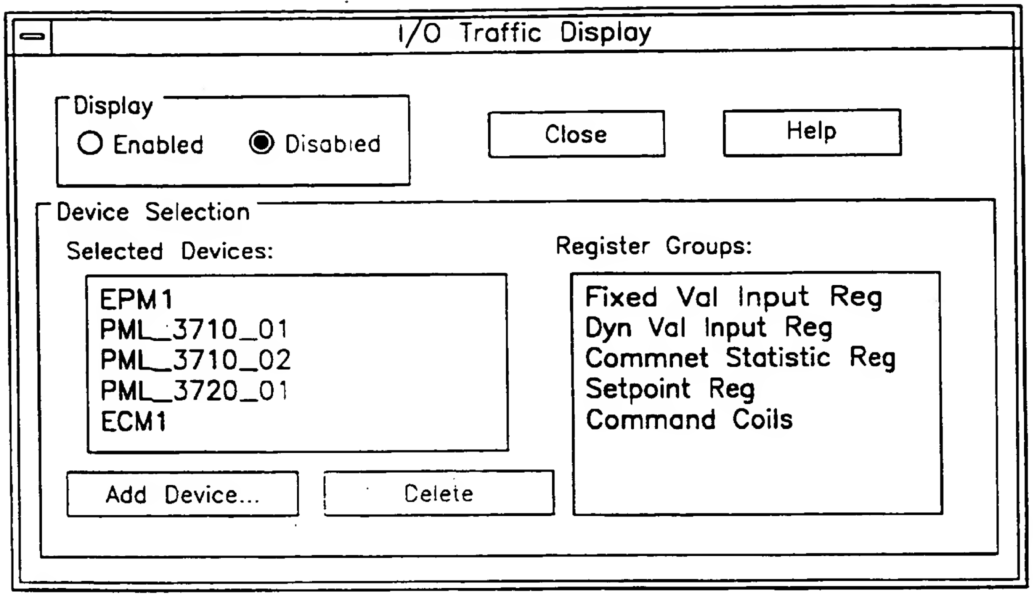

Server Windows Application–Server
Server
View
Configure
Help
I/O Traffic Display...
Clear Display

FIG. 59



Display

○ Enabled ● Disabled

Device Selection

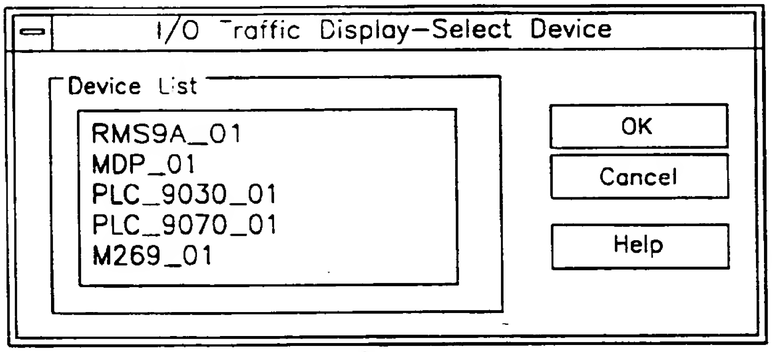
Selected Devices:

- EPM1
- PML_3710_01
- PML_3710_02
- PML_3720_01
- ECM1

Register Groups:

- Fixed Val Input Reg
- Dyn Val Input Reg
- Commnet Statistic Reg
- Setpoint Reg
- Command Coils

FIG. 60



Device List

- RMS9A_01
- MDP_01
- PLC_9030_01
- PLC_9070_01
- M269_01

FIG. 61

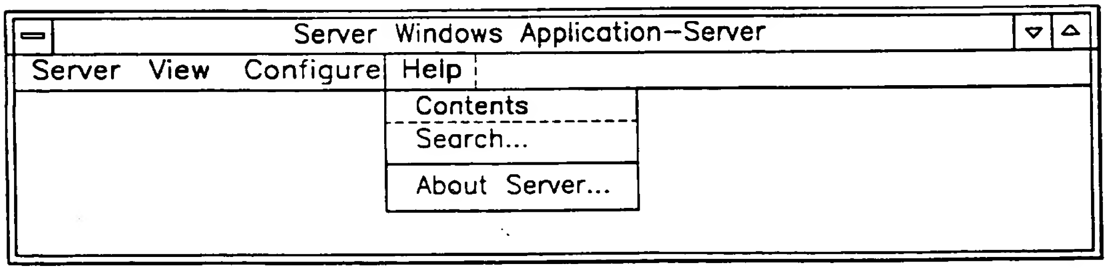


FIG. 62

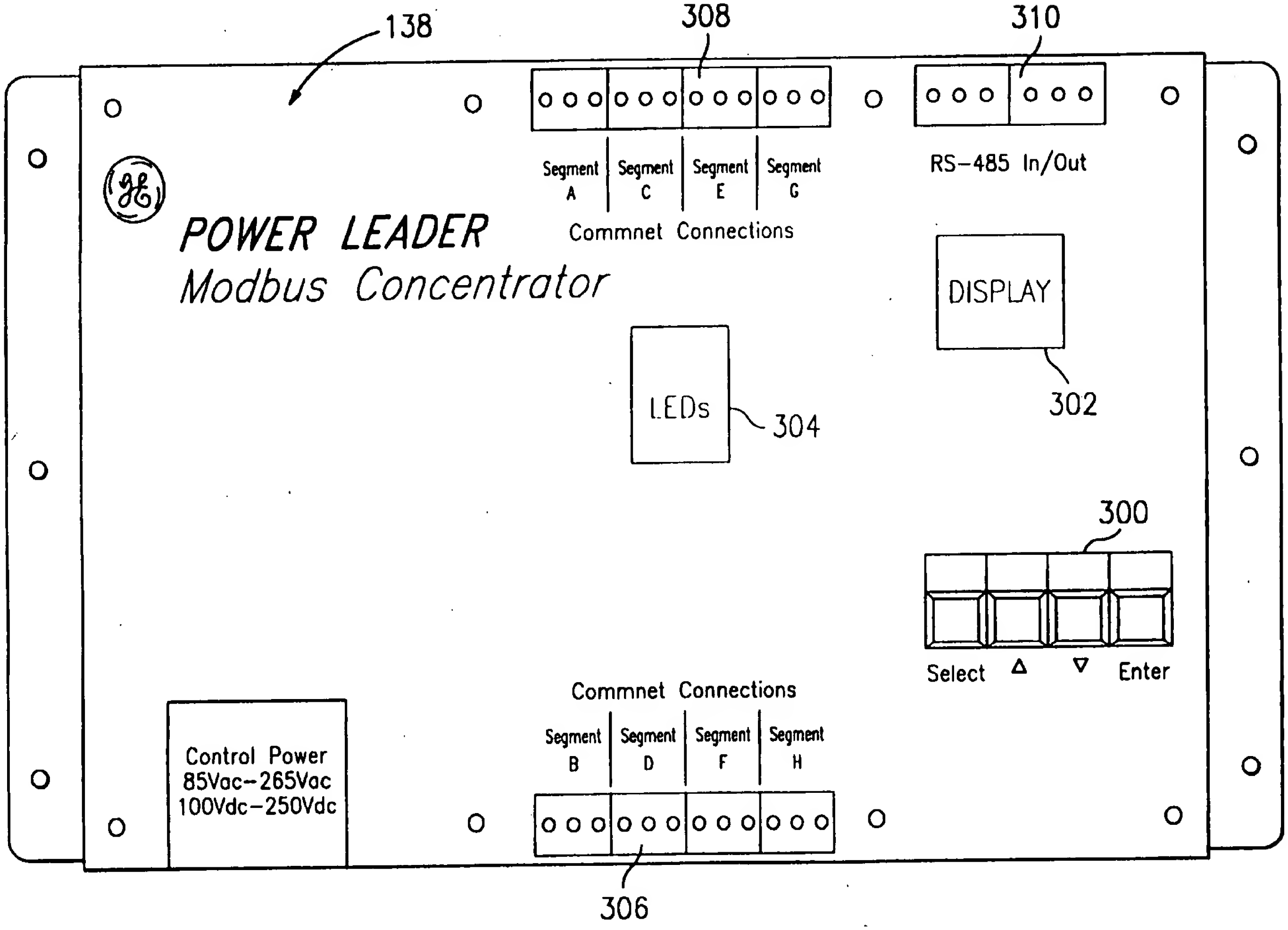
138
308
310
Segment A
Segment C
Segment E
Segment G
Commnet Connections
RS-485 In/Out
POWER LEADER
Modbus Concentrator
DISPLAY
302
LEDs
304
300
Select
Δ
∇
Enter
Commnet Connections
Segment B
Segment D
Segment F
Segment H
Control Power
85Vac-265Vac
100Vdc-250Vdc
306

FIG. 63

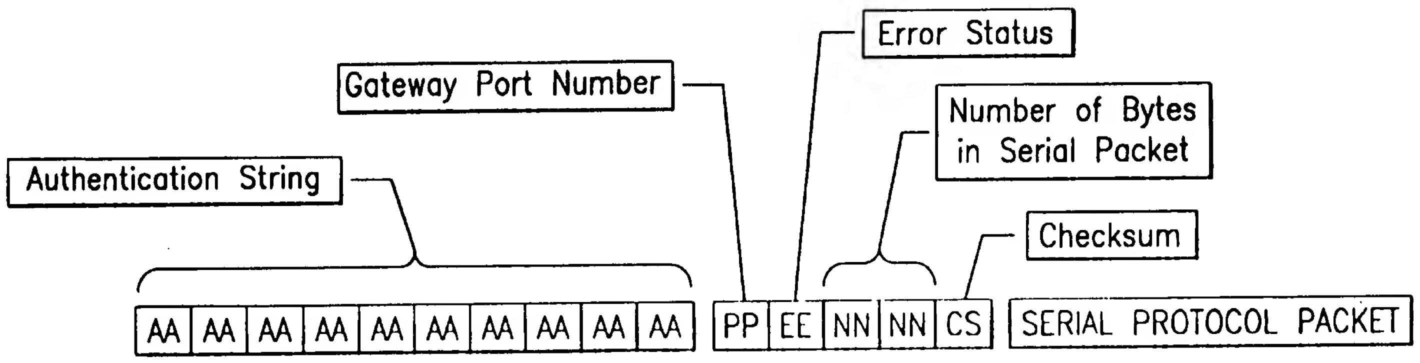

Error Status
Gateway Port Number
Number of Bytes in Serial Packet
Authentication String
Checksum
AA AA AA AA AA AA AA AA AA AA
PP EE NN NN CS
SERIAL PROTOCOL PACKET

FIG. 64

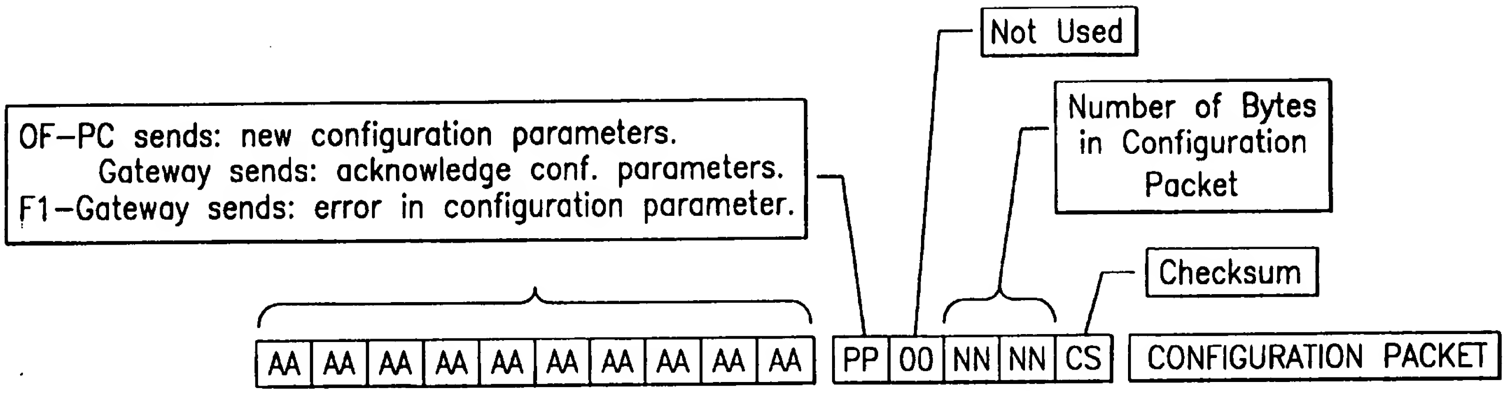

Not Used
Number of Bytes in Configuration Packet
OF–PC sends: new configuration parameters.
Gateway sends: acknowledge conf. parameters.
F1–Gateway sends: error in configuration parameter.
Checksum
AA AA AA AA AA AA AA AA AA AA
PP 00 NN NN CS
CONFIGURATION PACKET

FIG. 65

FIG. 66

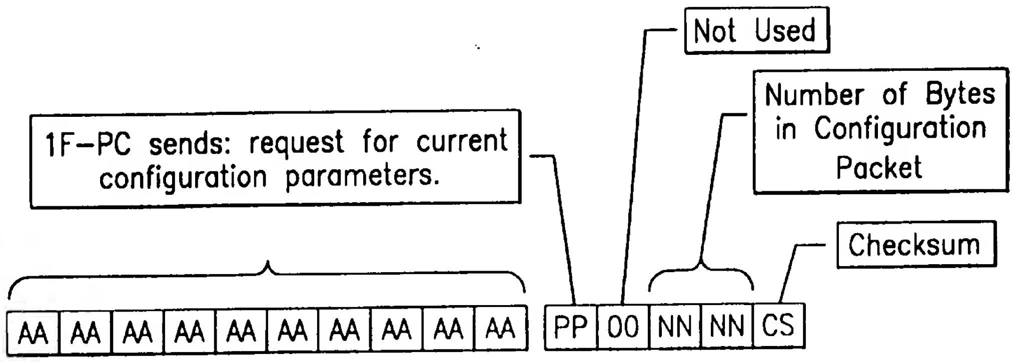
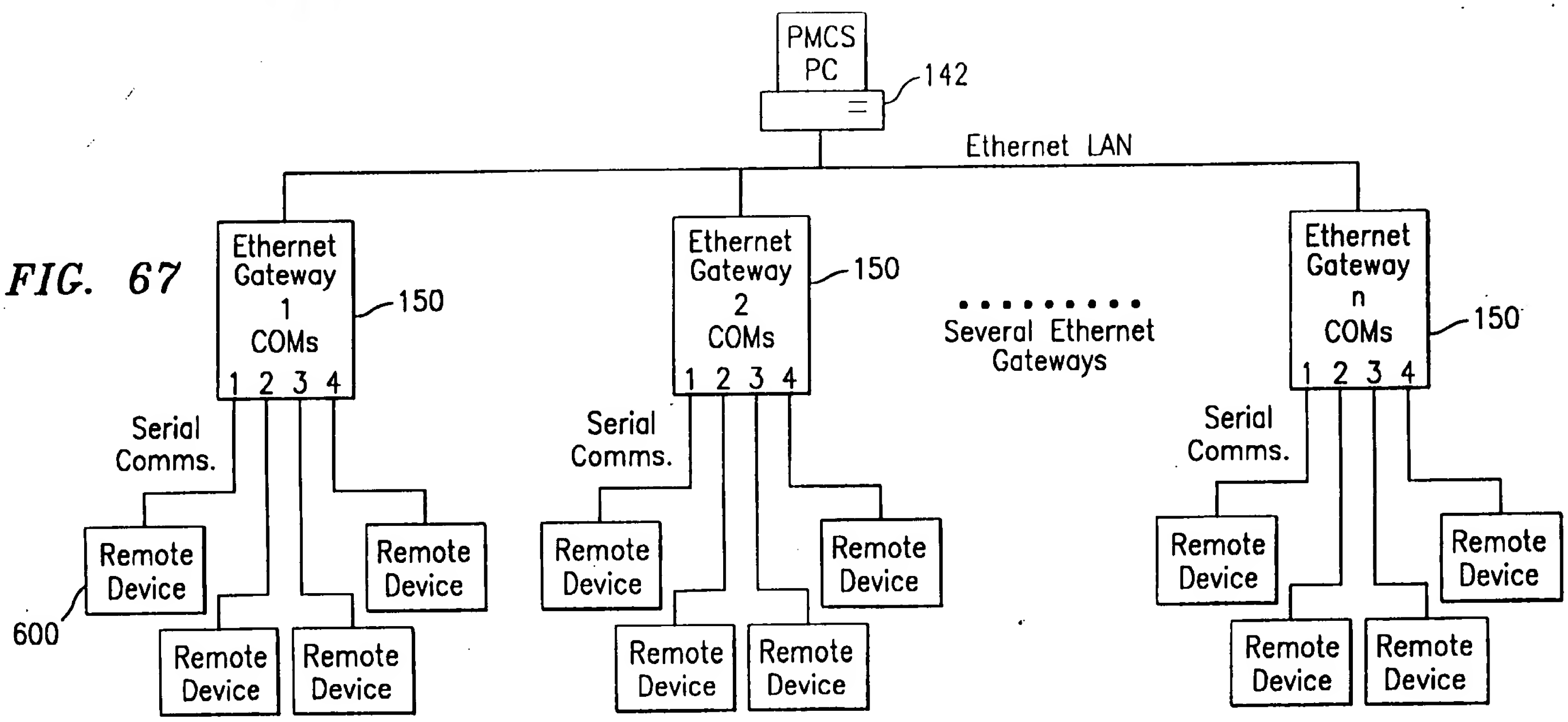

Not Used
Number of Bytes in Configuration Packet
1F–PC sends: request for current configuration parameters.
Checksum
AA
AA
AA
AA
AA
AA
AA
AA
AA
AA
PP
00
NN
NN
CS

FIG. 67


PMCS PC
142
Ethernet LAN
Ethernet Gateway 1 COMs
1 2 3 4
150
Ethernet Gateway 2 COMs
1 2 3 4
150
Several Ethernet Gateways
Ethernet Gateway n COMs
1 2 3 4
150
Serial Comms.
Serial Comms.
Serial Comms.
Remote Device
Remote Device
Remote Device
Remote Device
Remote Device
Remote Device
Remote Device
Remote Device
Remote Device
Remote Device
Remote Device
Remote Device
600

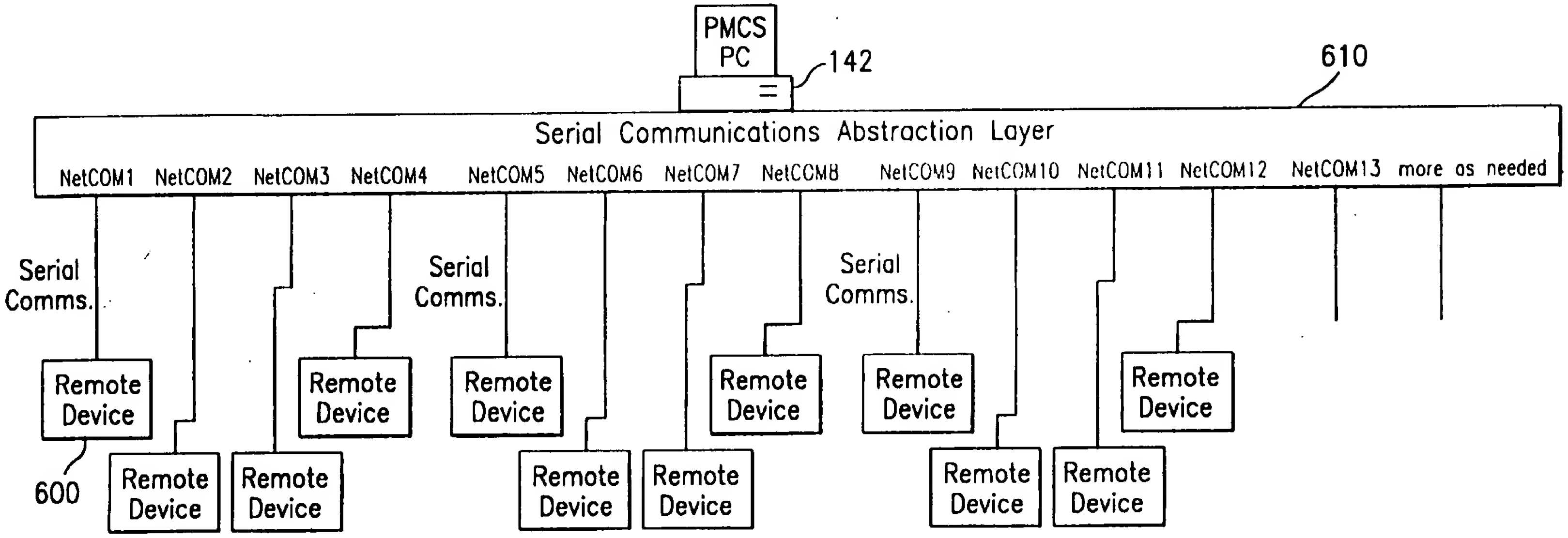

PMCS PC
142
610
Serial Communications Abstraction Layer
NetCOM1
NetCOM2
NetCOM3
NetCOM4
NetCOM5
NetCOM6
NetCOM7
NetCOM8
NetCOM9
NetCOM10
NetCOM11
NetCOM12
NetCOM13
more as needed
Serial Comms.
Serial Comms.
Serial Comms.
Remote Device
600

FIG. 68

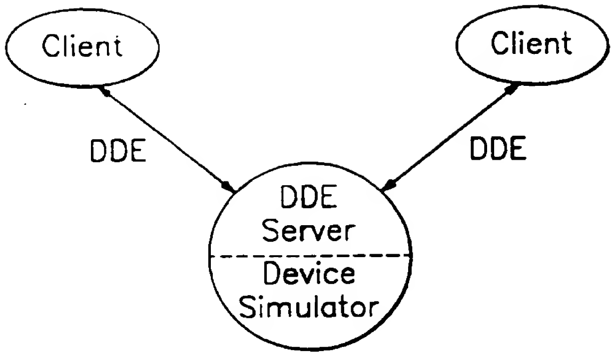
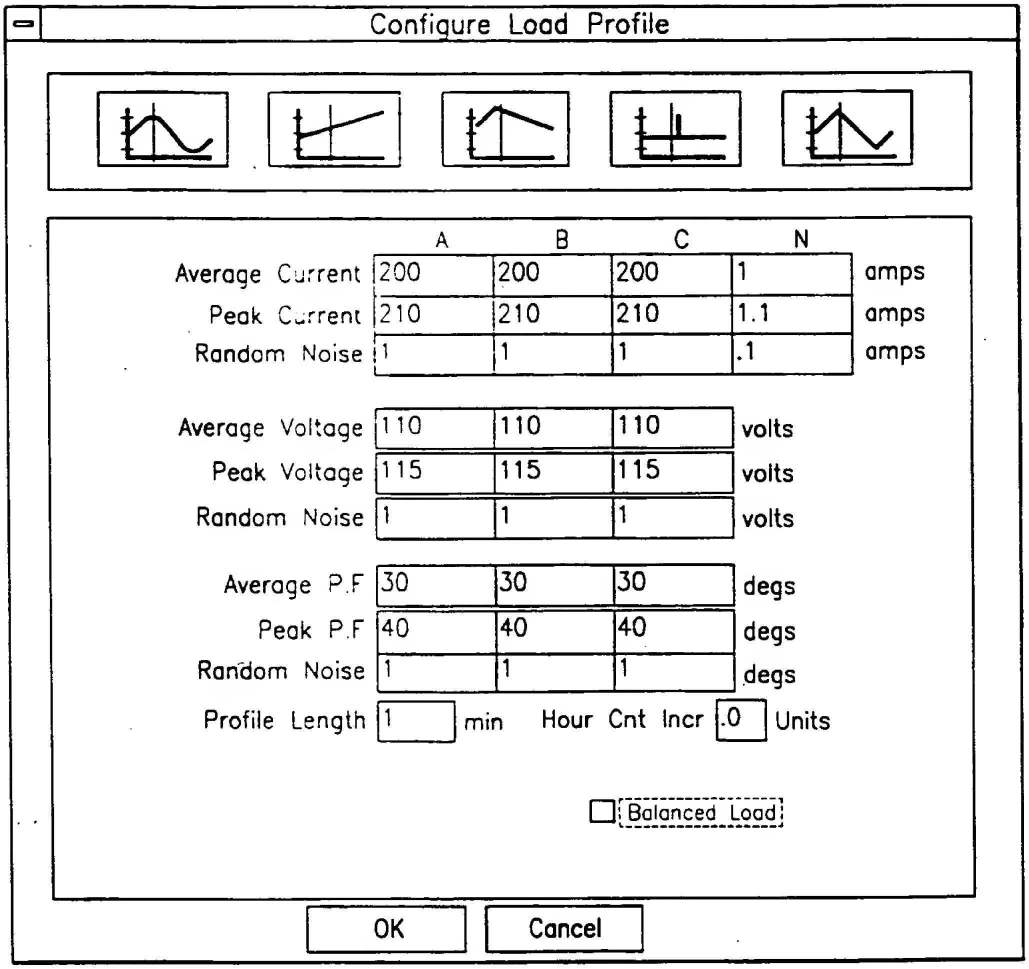


FIG. 69



Configure Load Profile

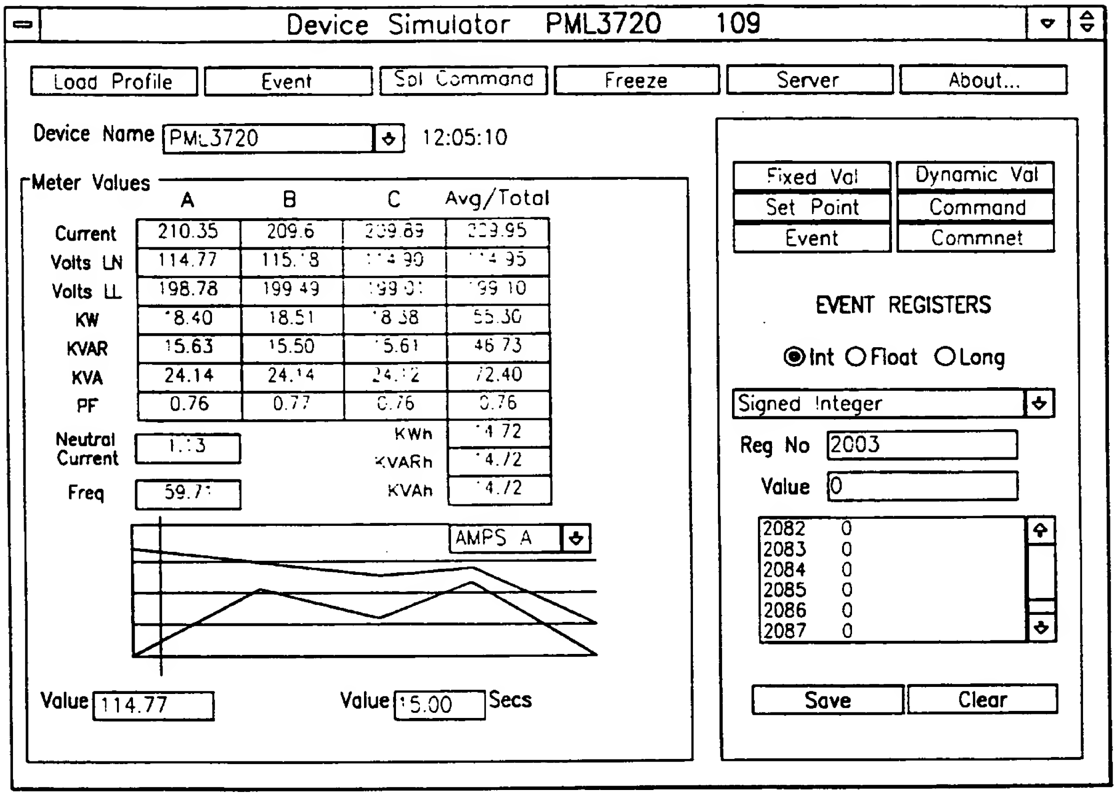
	A	B	C	N	
Average Current	200	200	200	1	amps
Peak Current	210	210	210	1.1	amps
Random Noise	1	1	1	.1	amps
Average Voltage	110	110	110		volts
Peak Voltage	115	115	115		volts
Random Noise	1	1	1		volts
Average P.F	30	30	30		degs
Peak P.F	40	40	40		degs
Random Noise	1	1	1		degs

Profile Length 1 min

Hour Cnt Incr .0 Units

☐ Balanced Load

FIG. 69A



Device Name PML3720 12:05:10

Meter Values

	A	B	C	Avg/Total
Current	210.35	209.6	209.89	209.95
Volts LN	114.77	115.18	114.90	114.95
Volts LL	198.78	199.49	199.01	199.10
KW	18.40	18.51	18.38	55.30
KVAR	15.63	15.50	15.61	46.73
KVA	24.14	24.14	24.12	72.40
PF	0.76	0.77	0.76	0.76

Neutral Current 1.13

Freq 59.71

KWh 14.72

KVARh 14.72

KVAh 14.72

Value 114.77

Value 15.00 Secs

EVENT REGISTERS

● Int ○ Float ○ Long

Signed Integer

Reg No 2003

Value 0

2082	0
2083	0
2084	0
2085	0
2086	0
2087	0

FIG. 69B

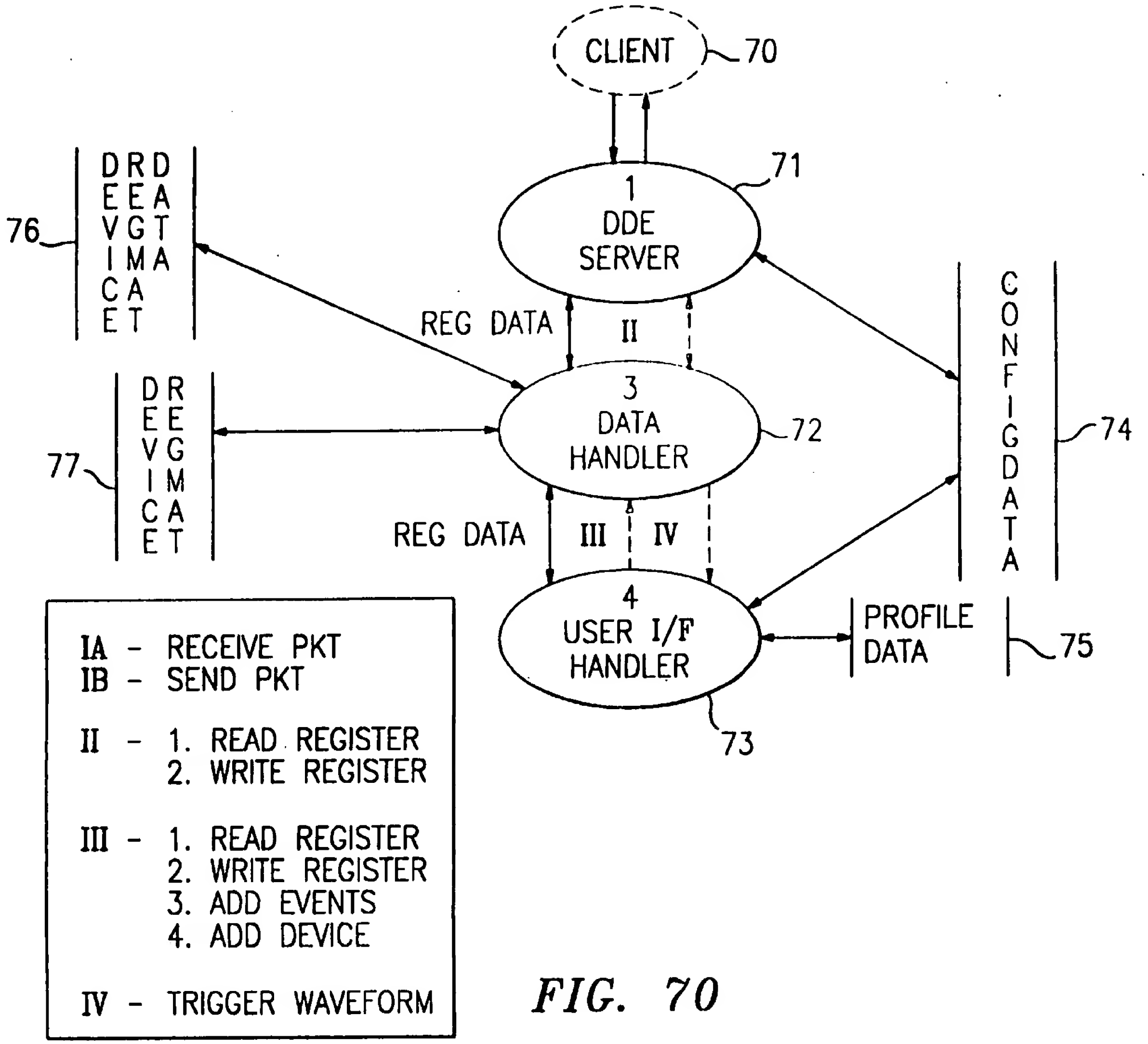

CLIENT
70
1
DDE
SERVER
71
DEVICE
REG DATA
76
REG DATA
II
3
DATA
HANDLER
72
DEVICE
REG MAP
77
REG DATA
III
IV
4
USER I/F
HANDLER
73
CONFIG DATA
74
PROFILE
DATA
75
IA - RECEIVE PKT
IB - SEND PKT
II - 1. READ REGISTER
2. WRITE REGISTER
III - 1. READ REGISTER
2. WRITE REGISTER
3. ADD EVENTS
4. ADD DEVICE
IV - TRIGGER WAVEFORM

FIG. 70

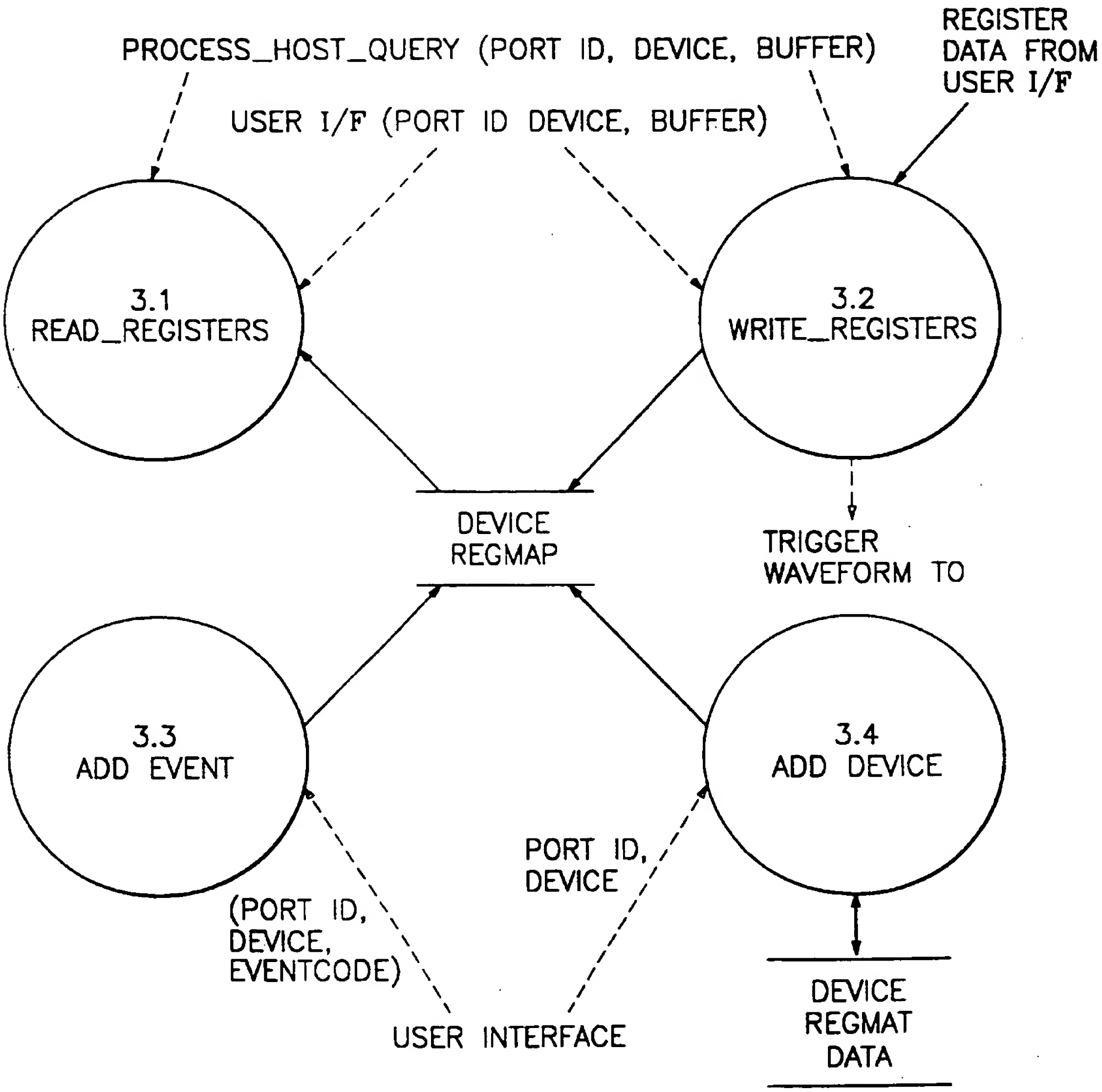

PROCESS_HOST_QUERY (PORT ID, DEVICE, BUFFER)
REGISTER DATA FROM USER I/F
USER I/F (PORT ID DEVICE, BUFFER)
3.1 READ_REGISTERS
3.2 WRITE_REGISTERS
DEVICE REGMAP
TRIGGER WAVEFORM TO
3.3 ADD EVENT
3.4 ADD DEVICE
PORT ID, DEVICE
(PORT ID, DEVICE, EVENTCODE)
USER INTERFACE
DEVICE REGMAT DATA

FIG. 71

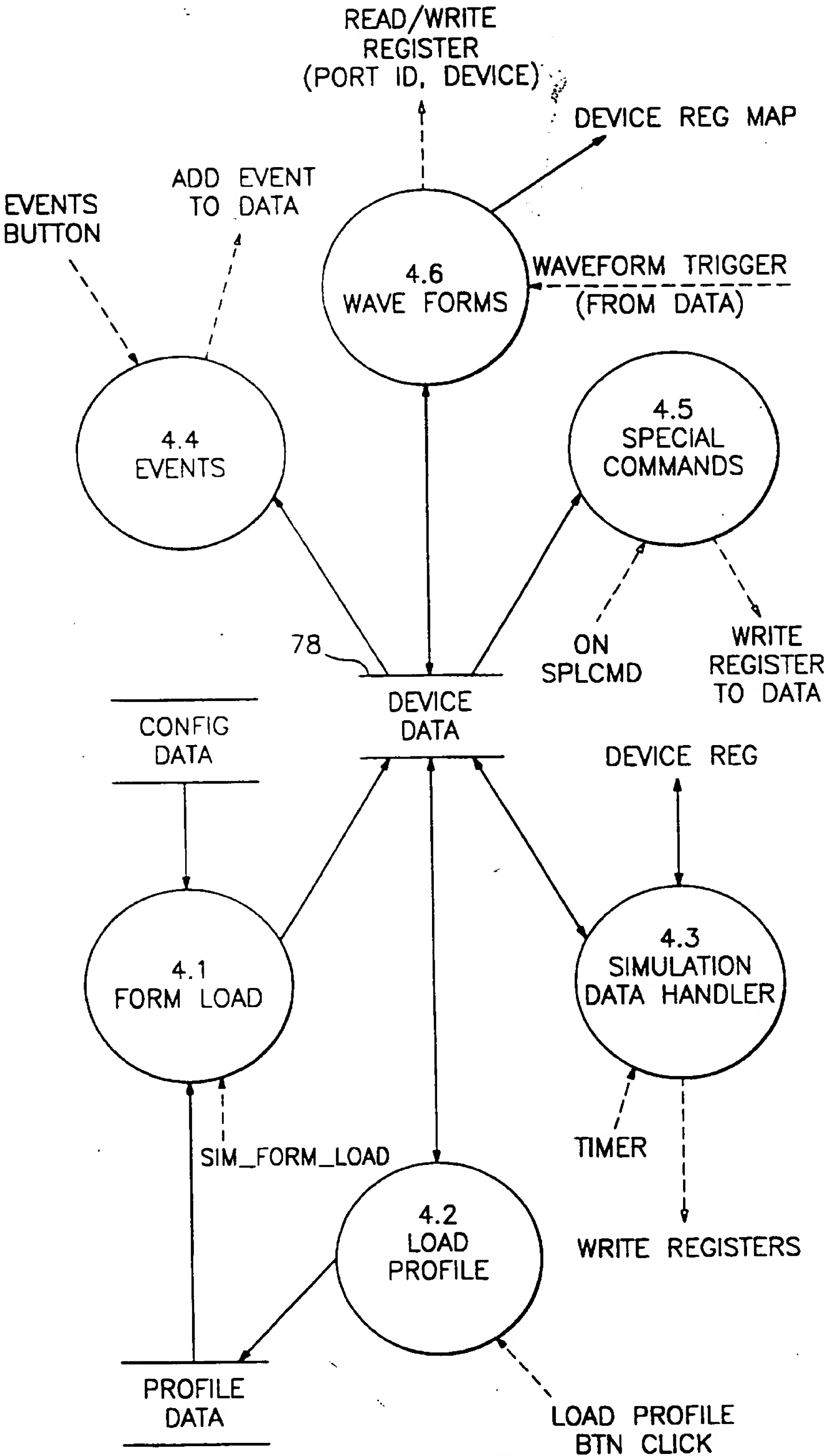

READ/WRITE REGISTER (PORT ID, DEVICE)
DEVICE REG MAP
EVENTS BUTTON
ADD EVENT TO DATA
4.6 WAVE FORMS
WAVEFORM TRIGGER (FROM DATA)
4.4 EVENTS
4.5 SPECIAL COMMANDS
ON SPLCMD
WRITE REGISTER TO DATA
78
DEVICE DATA
CONFIG DATA
DEVICE REG
4.1 FORM LOAD
4.3 SIMULATION DATA HANDLER
SIM_FORM_LOAD
TIMER
4.2 LOAD PROFILE
WRITE REGISTERS
PROFILE DATA
LOAD PROFILE BTN CLICK

FIG. 72

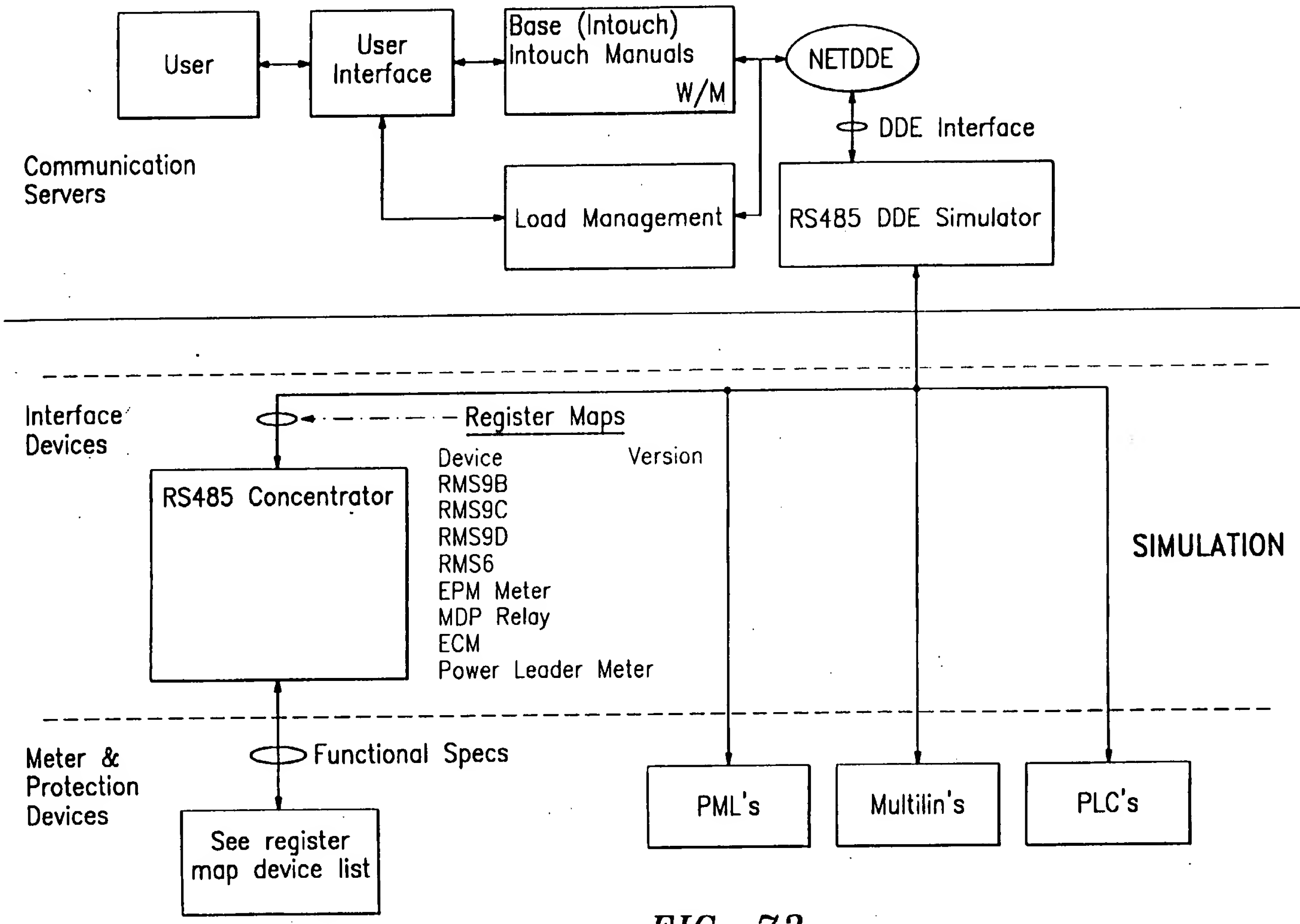

User
User Interface
Base (Intouch)
Intouch Manuals
W/M
NETDDE
DDE Interface
Load Management
RS485 DDE Simulator
Communication Servers
Interface Devices
Register Maps
Device
Version
RMS9B
RMS9C
RMS9D
RMS6
EPM Meter
MDP Relay
ECM
Power Leader Meter
RS485 Concentrator
SIMULATION
Meter & Protection Devices
Functional Specs
See register map device list
PML's
Multilin's
PLC's

FIG. 73

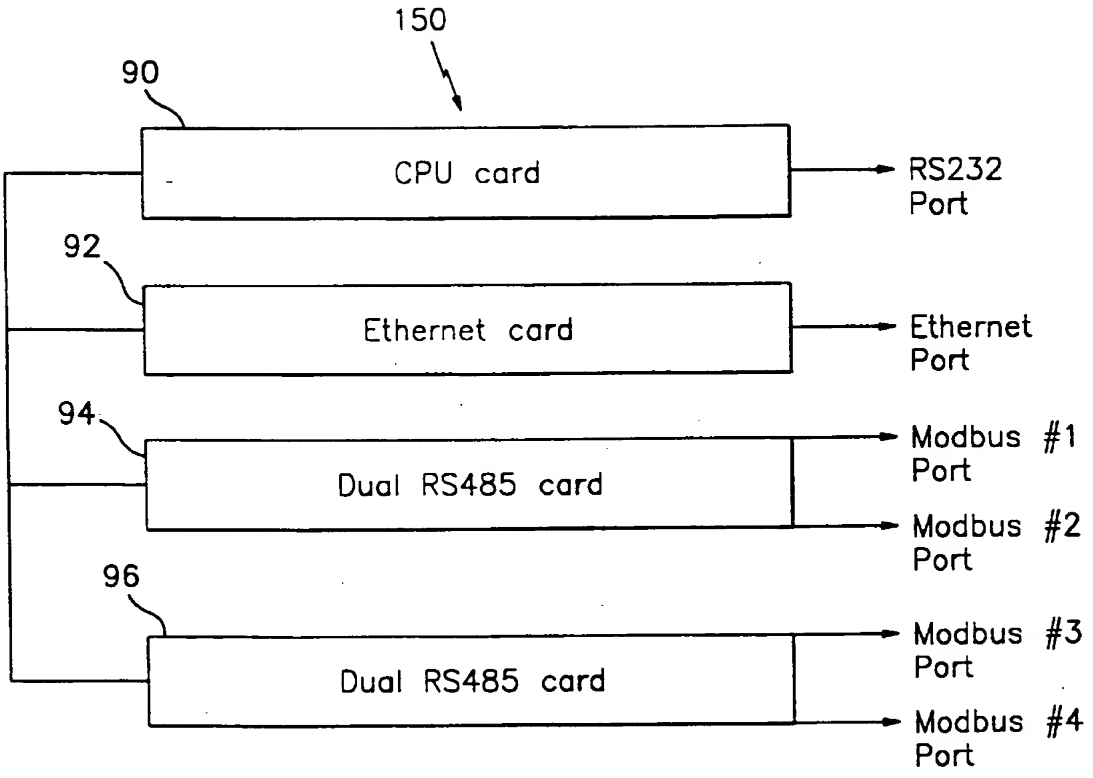

150
90
CPU card
RS232 Port
92
Ethernet card
Ethernet Port
94
Dual RS485 card
Modbus #1 Port
Modbus #2 Port
96
Dual RS485 card
Modbus #3 Port
Modbus #4 Port

FIG. 74

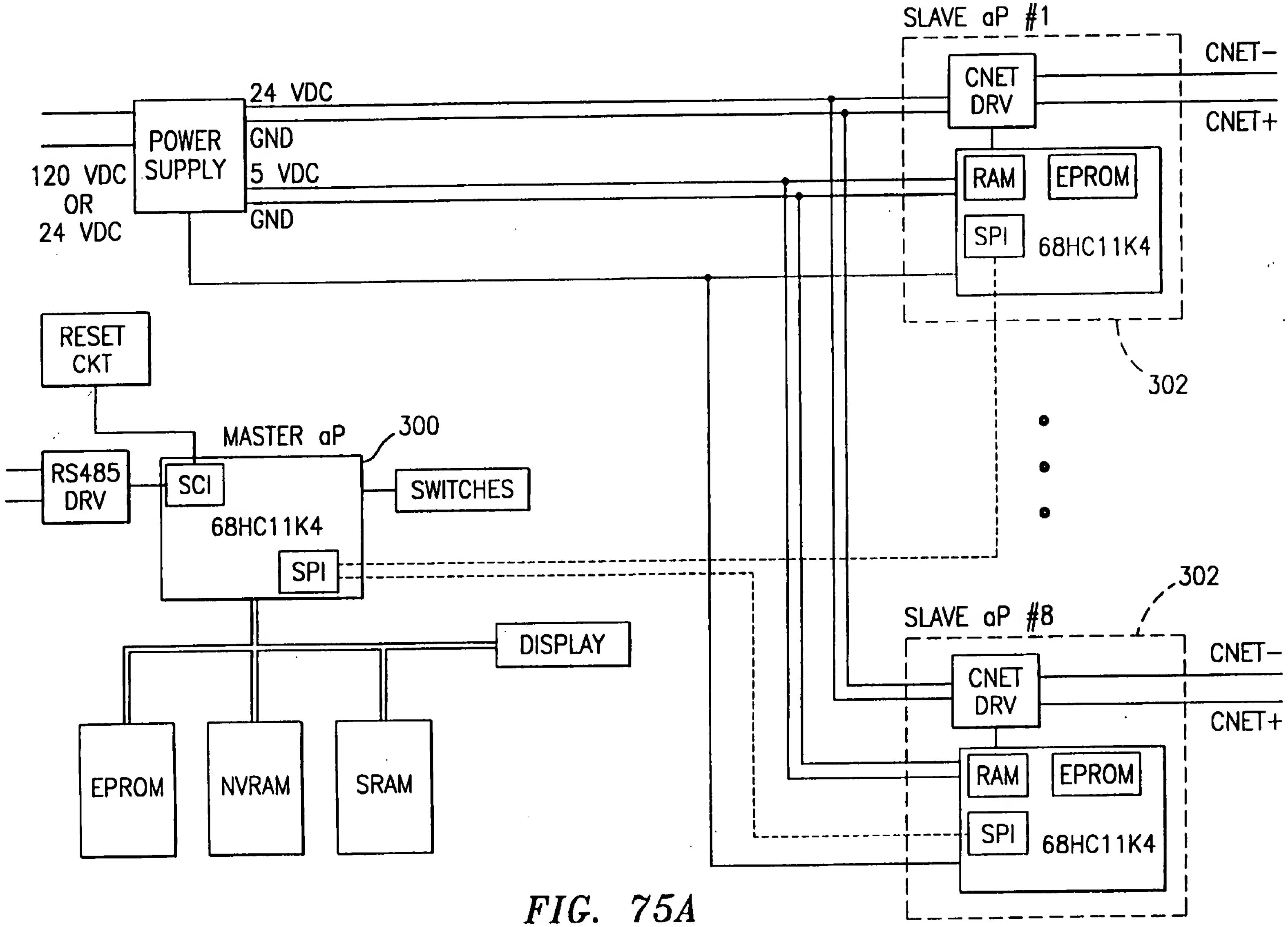

SLAVE aP #1
CNET DRV
CNET−
CNET+
RAM
EPROM
SPI
68HC11K4
302
24 VDC
GND
5 VDC
GND
POWER SUPPLY
120 VDC OR 24 VDC
RESET CKT
MASTER aP
300
RS485 DRV
SCI
68HC11K4
SPI
SWITCHES
DISPLAY
EPROM
NVRAM
SRAM
302
SLAVE aP #8
CNET DRV
CNET−
CNET+
RAM
EPROM
SPI
68HC11K4

FIG. 75A

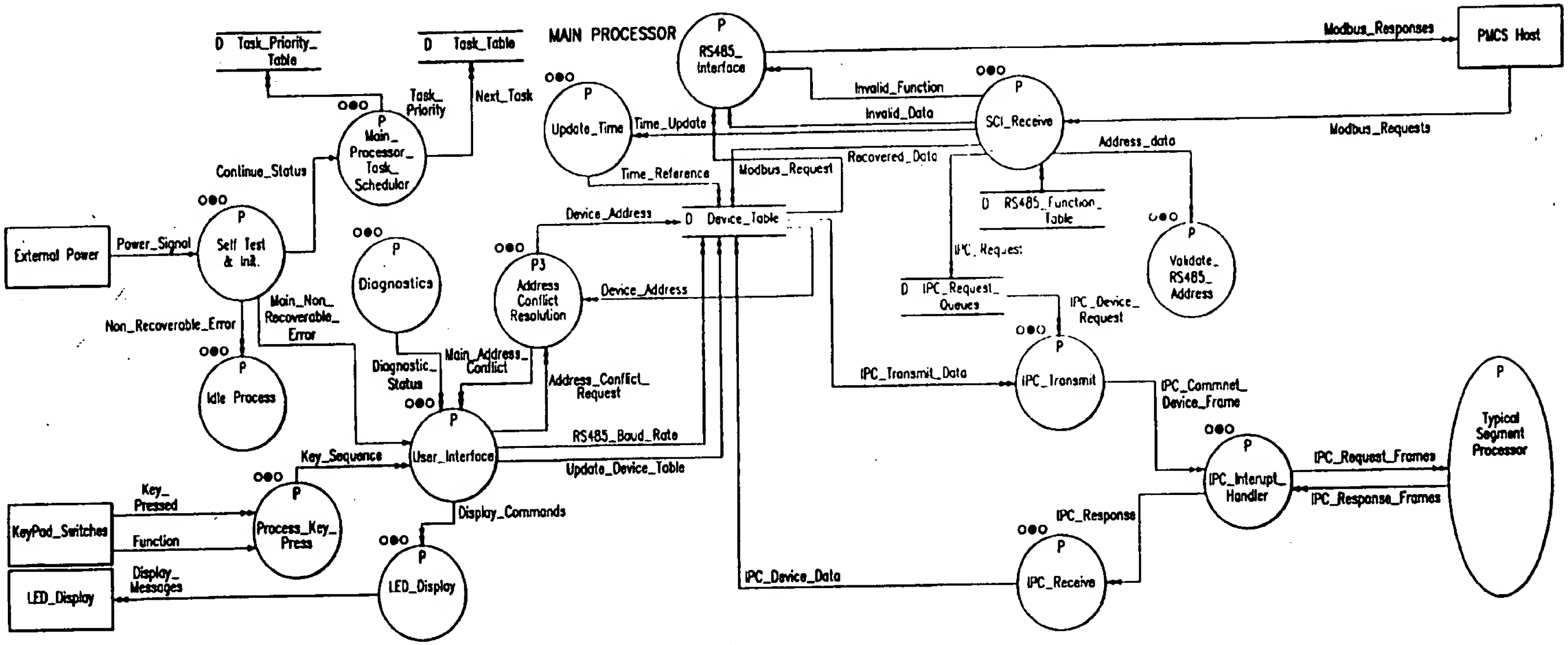

MAIN PROCESSOR
D Task_Priority_Table
D Task_Table
Task_Priority
Next_Task
P Main_Processor_Task_Scheduler
Continue_Status
External Power
Power_Signal
P Self Test & Init.
Non_Recoverable_Error
Main_Non_Recoverable_Error
P Idle Process
P Diagnostics
Diagnostic_Status
Main_Address_Conflict
P User_Interface
Key_Sequence
KeyPad_Switches
Key_Pressed
Function
P Process_Key_Press
LED_Display
Display_Messages
P LED_Display
Display_Commands
P Update_Time
Time_Update
Time_Reference
P RS485_Interface
Modbus_Request
Device_Address
D Device_Table
P3 Address Conflict Resolution
Device_Address
Address_Conflict_Request
RS485_Baud_Rate
Update_Device_Table
Invalid_Function
Invalid_Data
Recovered_Data
P SCI_Receive
Modbus_Responses
Modbus_Requests
PMCS Host
Address_data
D RS485_Function_Table
P Validate_RS485_Address
IPC_Request
D IPC_Request_Queues
IPC_Device_Request
P IPC_Transmit
IPC_Transmit_Data
IPC_Commnet_Device_Frame
P IPC_Interupt_Handler
IPC_Request_Frames
IPC_Response_Frames
P Typical Segment Processor
IPC_Response
P IPC_Receive
IPC_Device_Data

FIG. 75B

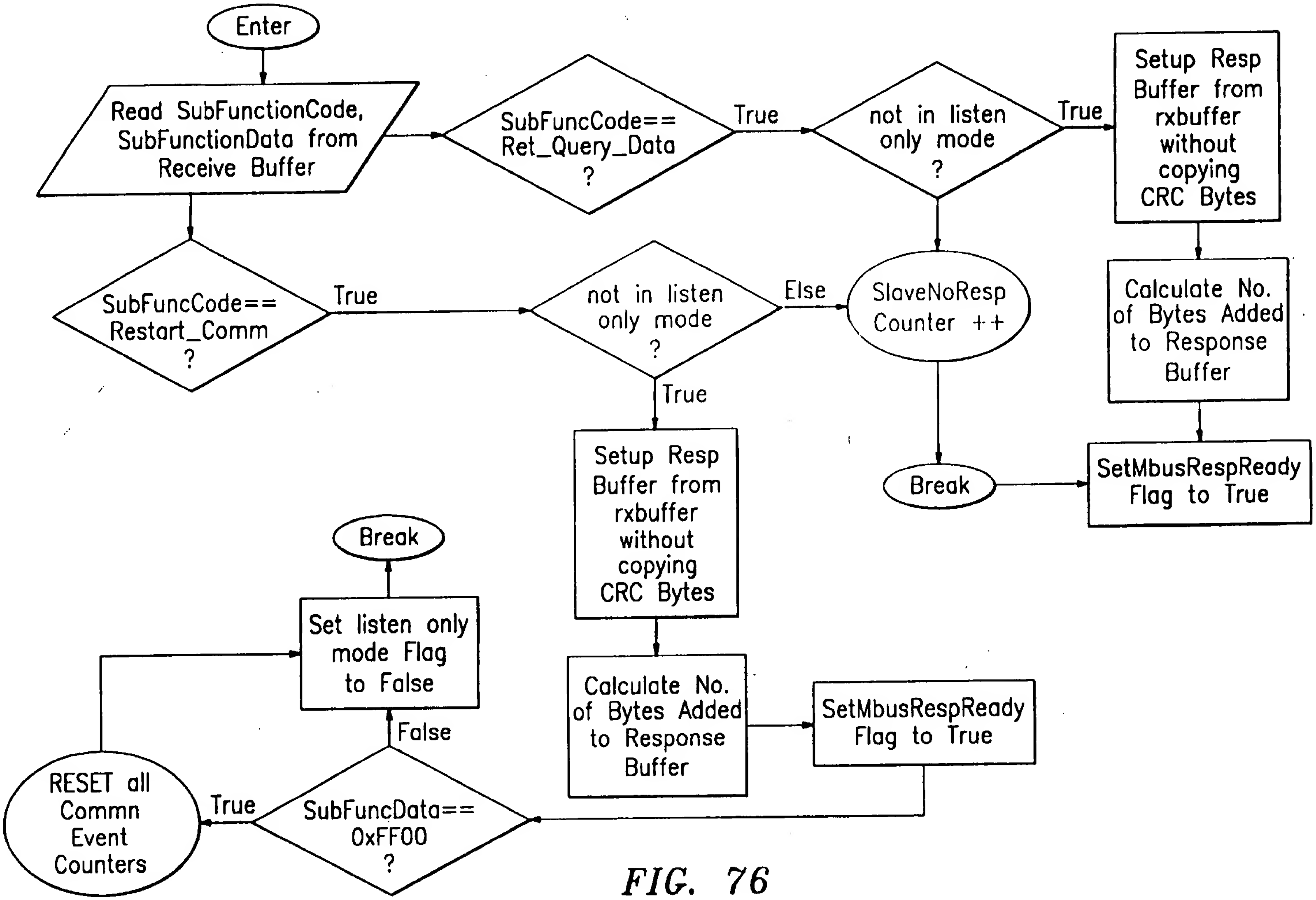

Enter
Read SubFunctionCode, SubFunctionData from Receive Buffer
SubFuncCode== Ret_Query_Data ?
True
not in listen only mode ?
True
Setup Resp Buffer from rxbuffer without copying CRC Bytes
Calculate No. of Bytes Added to Response Buffer
SetMbusRespReady Flag to True
SlaveNoResp Counter ++
Break
SubFuncCode== Restart_Comm ?
True
not in listen only mode ?
Else
True
Setup Resp Buffer from rxbuffer without copying CRC Bytes
Calculate No. of Bytes Added to Response Buffer
SetMbusRespReady Flag to True
SubFuncData== 0xFF00 ?
True
False
RESET all Commn Event Counters
Set listen only mode Flag to False
Break

FIG. 76

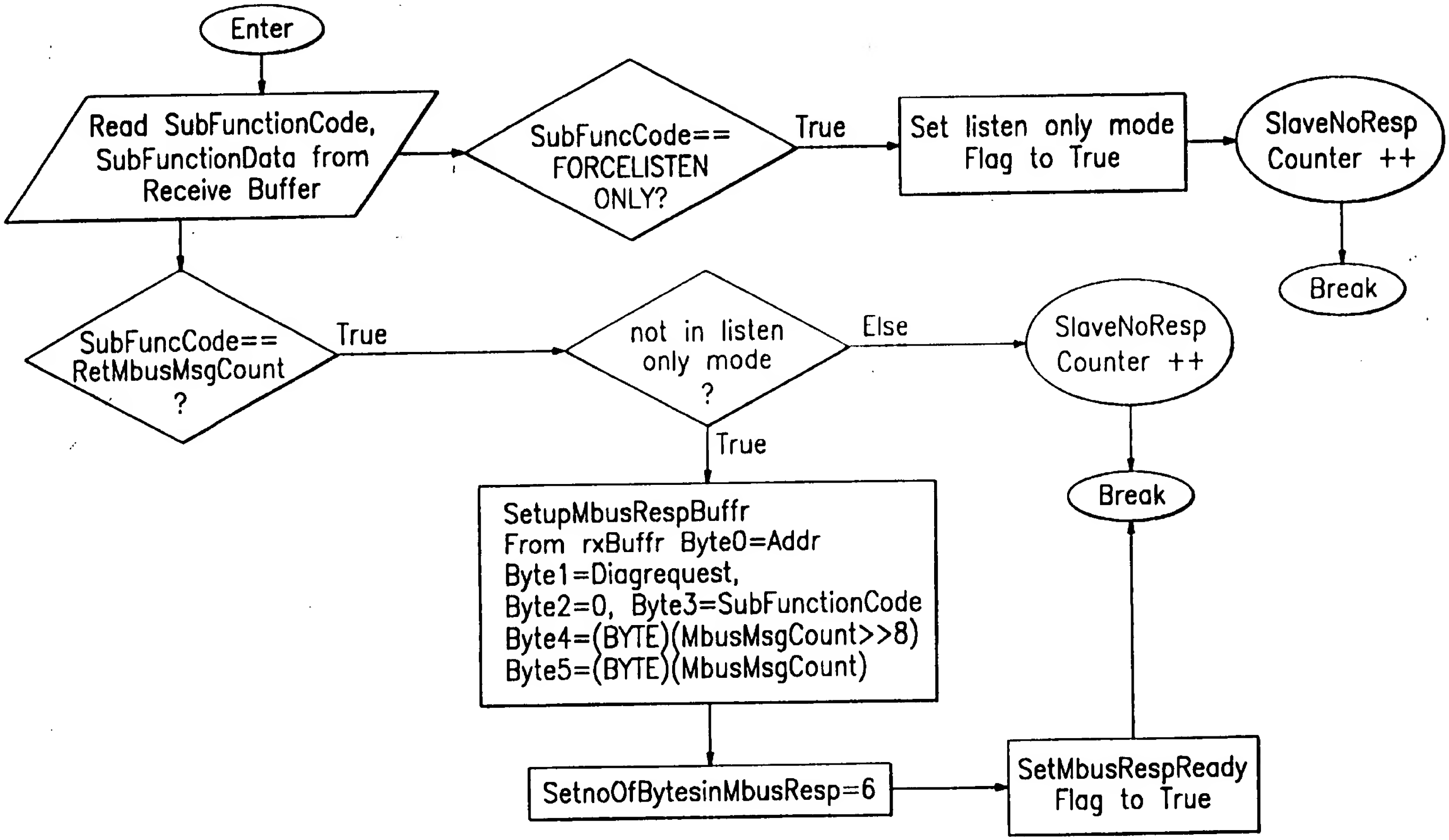

Enter
Read SubFunctionCode, SubFunctionData from Receive Buffer
SubFuncCode== FORCELISTEN ONLY?
True
Set listen only mode Flag to True
SlaveNoResp Counter ++
Break
SubFuncCode== RetMbusMsgCount ?
True
not in listen only mode ?
Else
SlaveNoResp Counter ++
Break
True
SetupMbusRespBuffr
From rxBuffr Byte0=Addr
Byte1=Diagrequest,
Byte2=0, Byte3=SubFunctionCode
Byte4=(BYTE)(MbusMsgCount>>8)
Byte5=(BYTE)(MbusMsgCount)
SetnoOfBytesinMbusResp=6
SetMbusRespReady Flag to True

FIG. 77

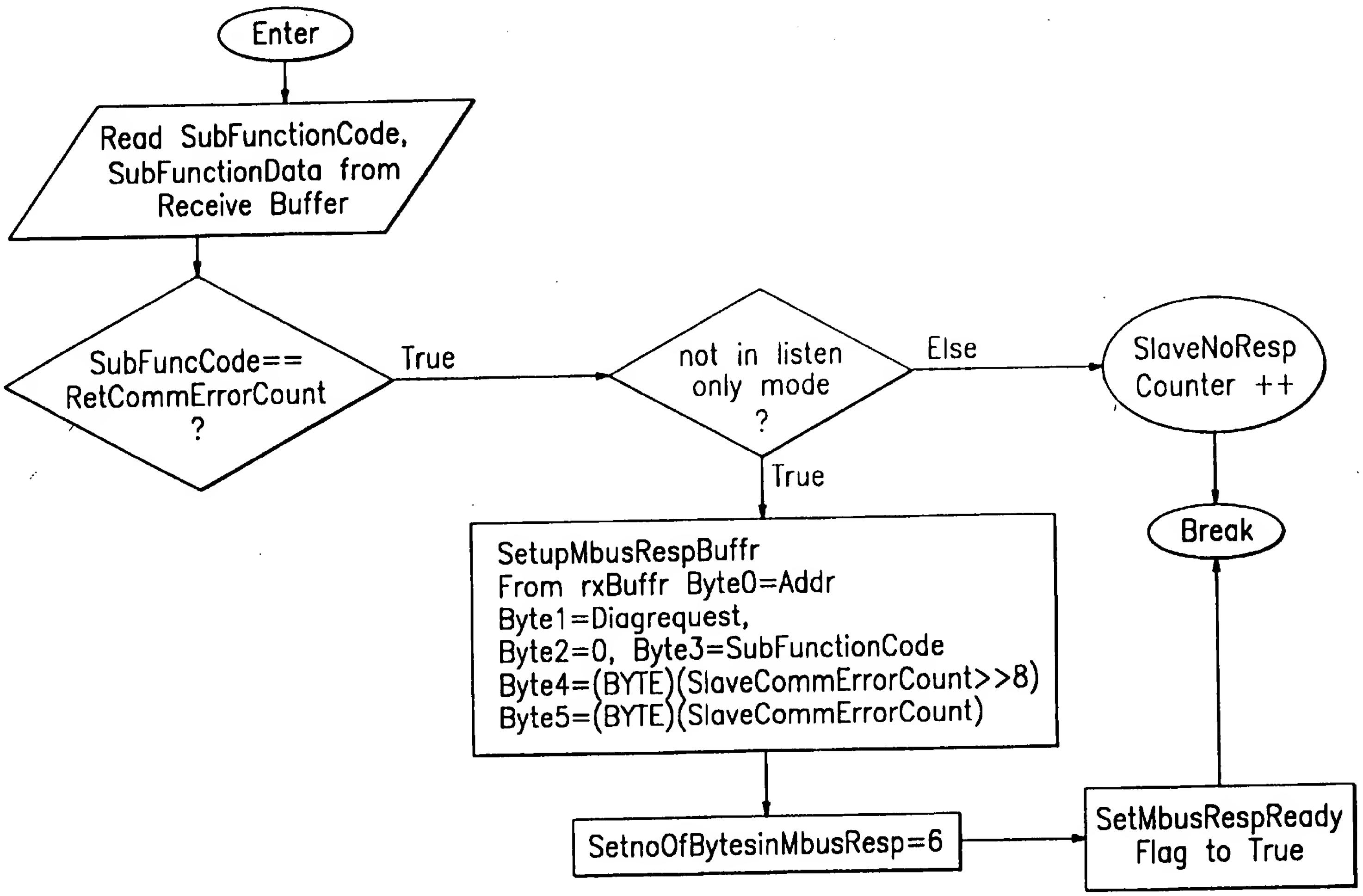

Enter
Read SubFunctionCode, SubFunctionData from Receive Buffer
SubFuncCode== RetCommErrorCount ?
True
not in listen only mode ?
Else
SlaveNoResp Counter ++
True
SetupMbusRespBuffr
From rxBuffr Byte0=Addr
Byte1=Diagrequest,
Byte2=0, Byte3=SubFunctionCode
Byte4=(BYTE)(SlaveCommErrorCount>>8)
Byte5=(BYTE)(SlaveCommErrorCount)
Break
SetnoOfBytesinMbusResp=6
SetMbusRespReady Flag to True

FIG. 78

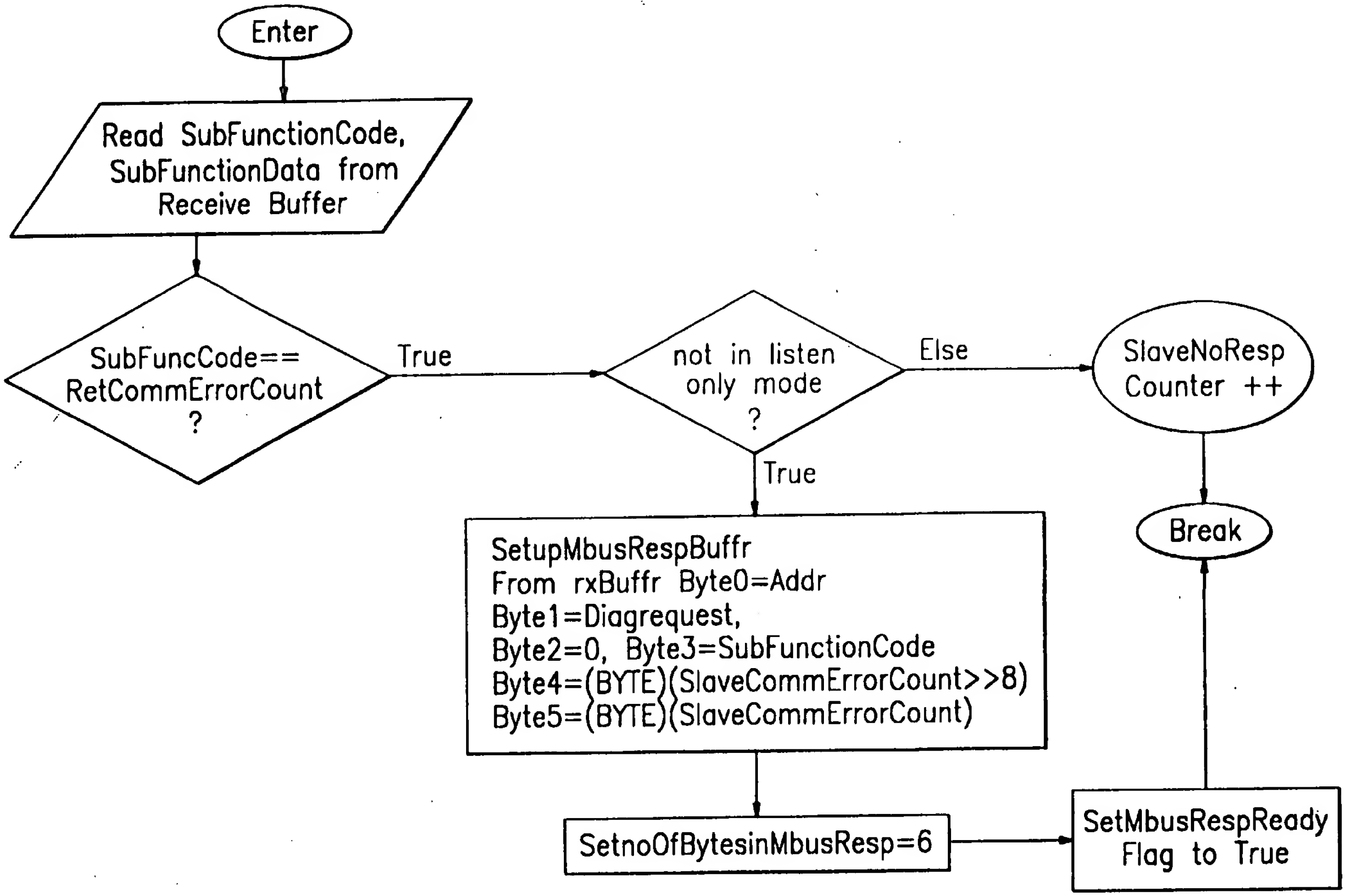

Enter
Read SubFunctionCode, SubFunctionData from Receive Buffer
SubFuncCode== RetCommErrorCount ?
True
not in listen only mode ?
Else
SlaveNoResp Counter ++
True
SetupMbusRespBuffr
From rxBuffr Byte0=Addr
Byte1=Diagrequest,
Byte2=0, Byte3=SubFunctionCode
Byte4=(BYTE)(SlaveCommErrorCount>>8)
Byte5=(BYTE)(SlaveCommErrorCount)
Break
SetnoOfBytesinMbusResp=6
SetMbusRespReady Flag to True

FIG. 79

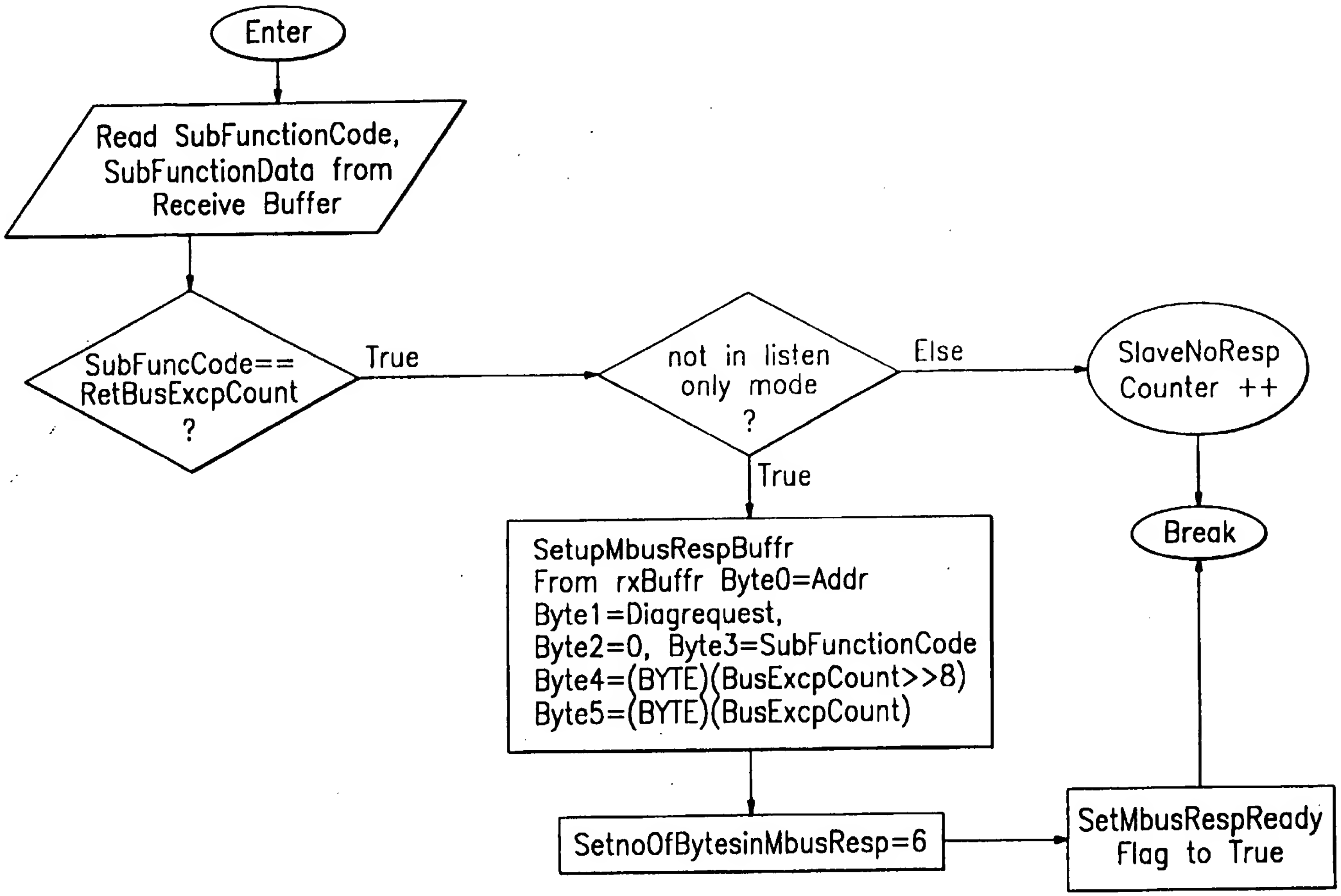

Enter
Read SubFunctionCode, SubFunctionData from Receive Buffer
SubFuncCode== RetBusExcpCount ?
True
not in listen only mode ?
Else
SlaveNoResp Counter ++
True
SetupMbusRespBuffr
From rxBuffr Byte0=Addr
Byte1=Diagrequest,
Byte2=0, Byte3=SubFunctionCode
Byte4=(BYTE)(BusExcpCount>>8)
Byte5=(BYTE)(BusExcpCount)
Break
SetnoOfBytesinMbusResp=6
SetMbusRespReady Flag to True

FIG. 80

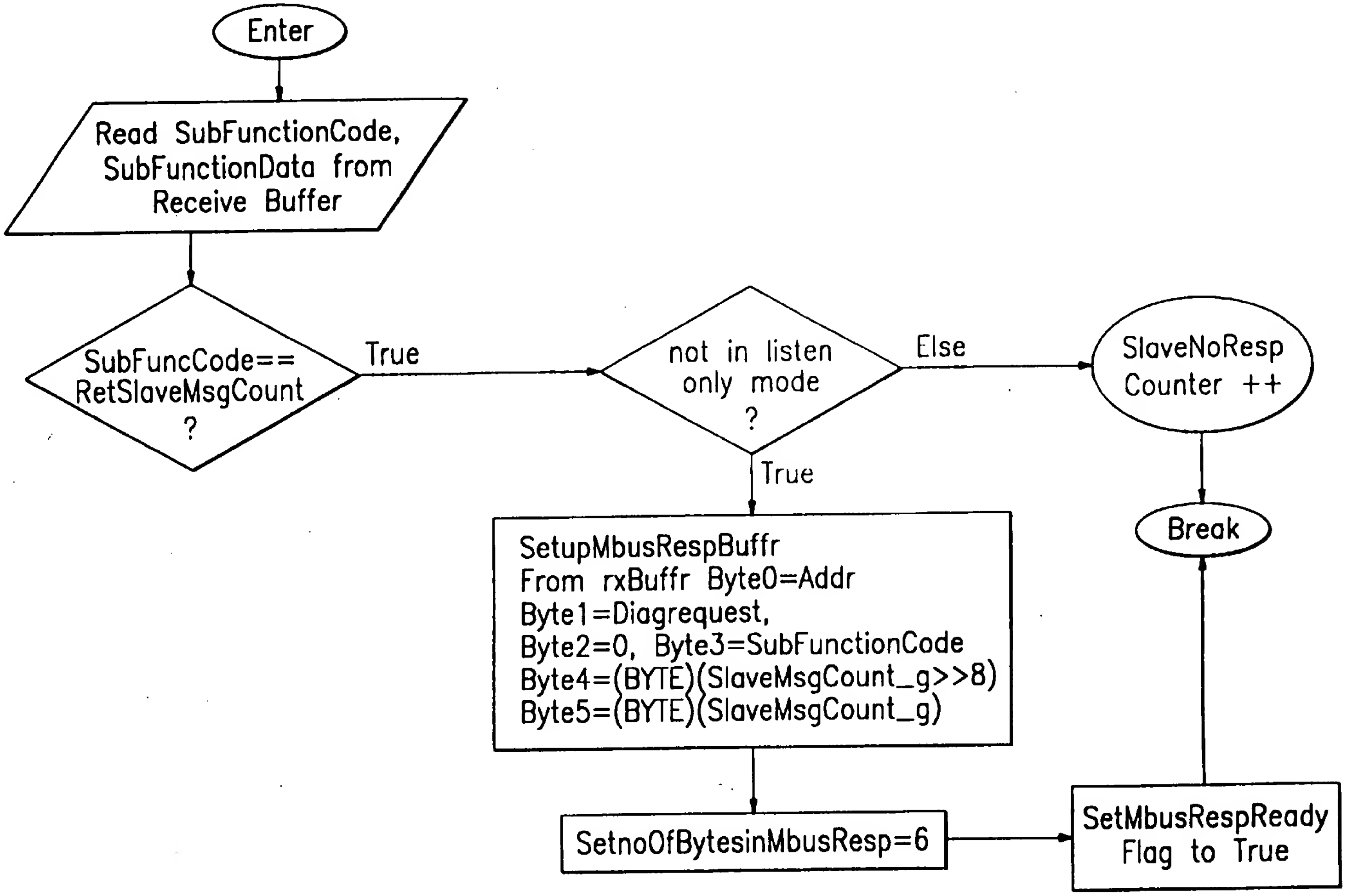

Enter
Read SubFunctionCode, SubFunctionData from Receive Buffer
SubFuncCode== RetSlaveMsgCount ?
True
not in listen only mode ?
Else
SlaveNoResp Counter ++
True
SetupMbusRespBuffr
From rxBuffr Byte0=Addr
Byte1=Diagrequest,
Byte2=0, Byte3=SubFunctionCode
Byte4=(BYTE)(SlaveMsgCount_g>>8)
Byte5=(BYTE)(SlaveMsgCount_g)
SetnoOfBytesinMbusResp=6
SetMbusRespReady Flag to True
Break

FIG. 81

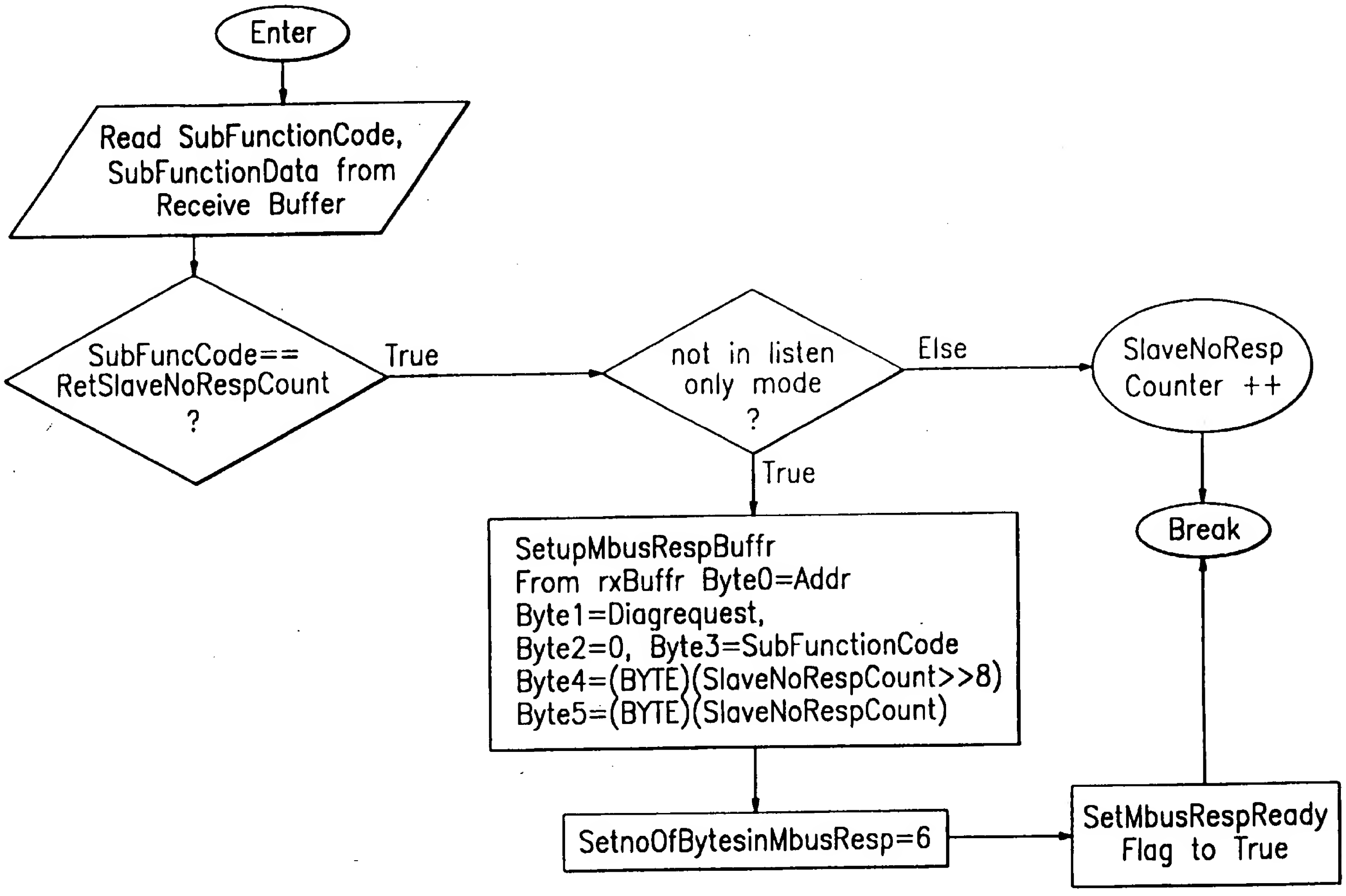

Enter
Read SubFunctionCode, SubFunctionData from Receive Buffer
SubFuncCode== RetSlaveNoRespCount ?
True
not in listen only mode ?
Else
SlaveNoResp Counter ++
True
SetupMbusRespBuffr
From rxBuffr Byte0=Addr
Byte1=Diagrequest,
Byte2=0, Byte3=SubFunctionCode
Byte4=(BYTE)(SlaveNoRespCount>>8)
Byte5=(BYTE)(SlaveNoRespCount)
Break
SetnoOfBytesinMbusResp=6
SetMbusRespReady Flag to True

FIG. 82

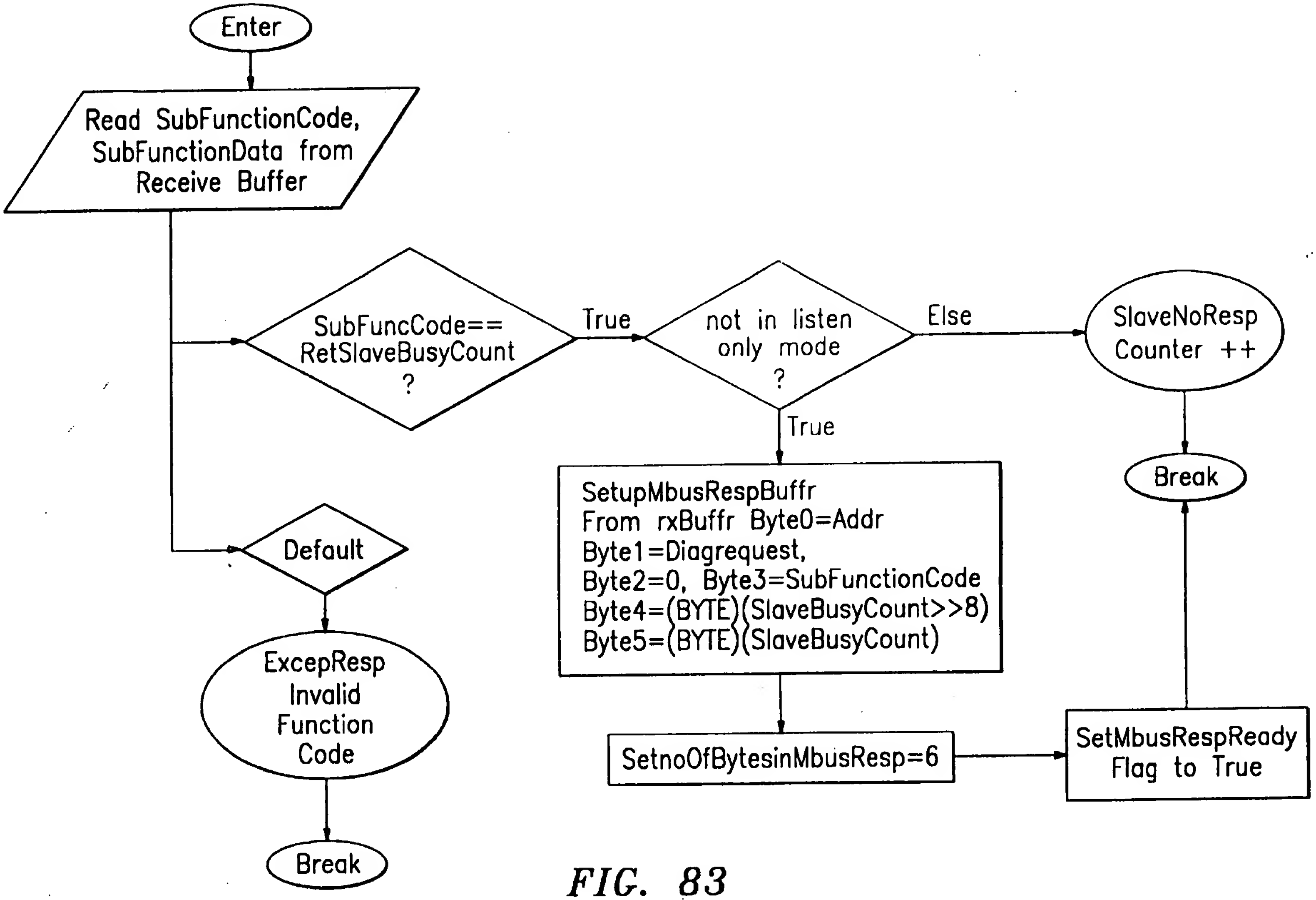
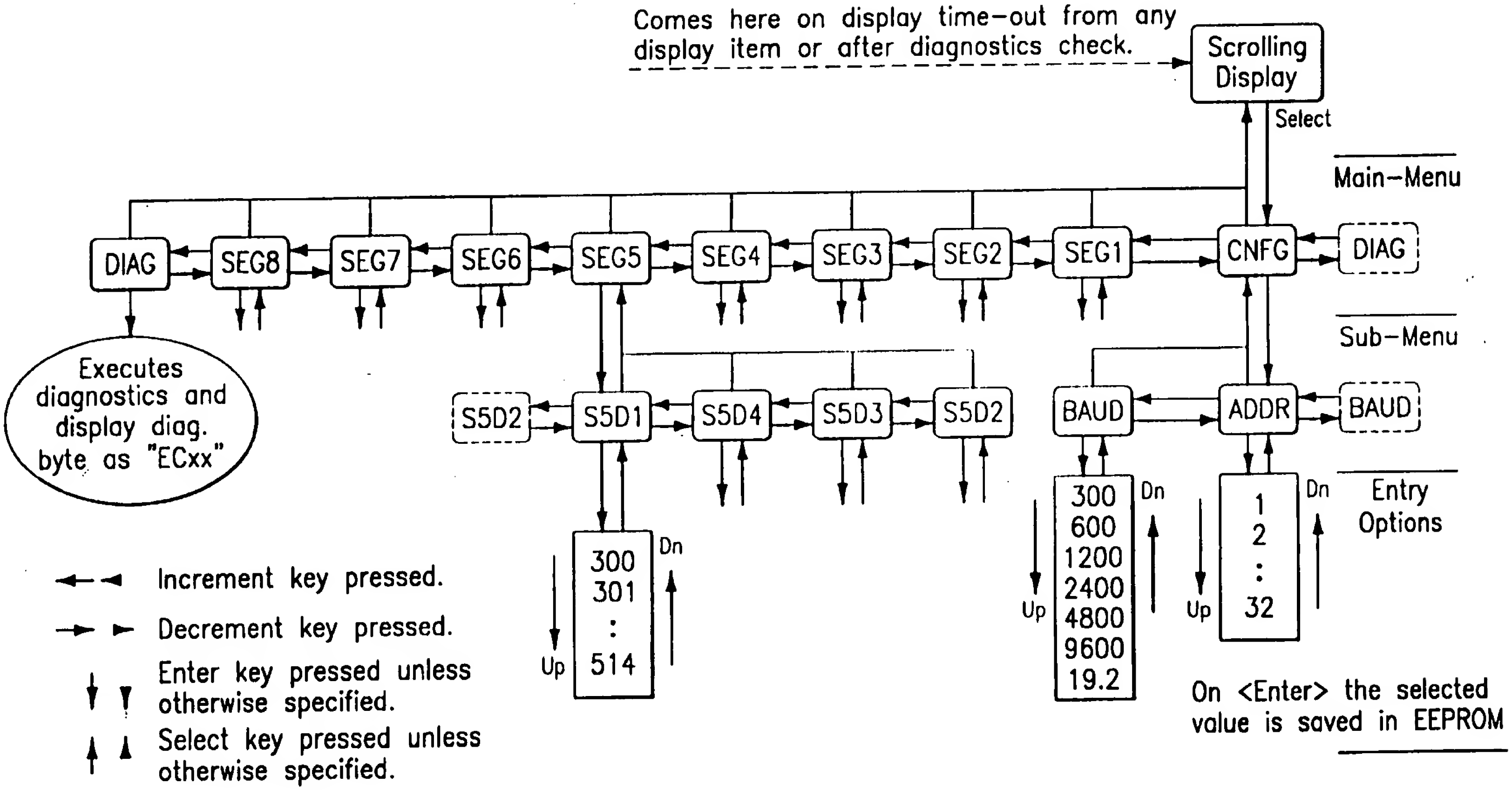

Enter
Read SubFunctionCode, SubFunctionData from Receive Buffer
SubFuncCode== RetSlaveBusyCount ?
True
not in listen only mode ?
Else
SlaveNoResp Counter ++
Break
True
SetupMbusRespBuffr
From rxBuffr Byte0=Addr
Byte1=Diagrequest,
Byte2=0, Byte3=SubFunctionCode
Byte4=(BYTE)(SlaveBusyCount>>8)
Byte5=(BYTE)(SlaveBusyCount)
SetnoOfBytesinMbusResp=6
SetMbusRespReady Flag to True
Default
ExcepResp Invalid Function Code
Break

FIG. 83



←◄ Increment key pressed.

→► Decrement key pressed.

↓▼ Enter key pressed unless otherwise specified.

↑▲ Select key pressed unless otherwise specified.

Display item S5D1 means device one of segment five. For other segment sub-menu shall be made accordingly.

FIG. 84

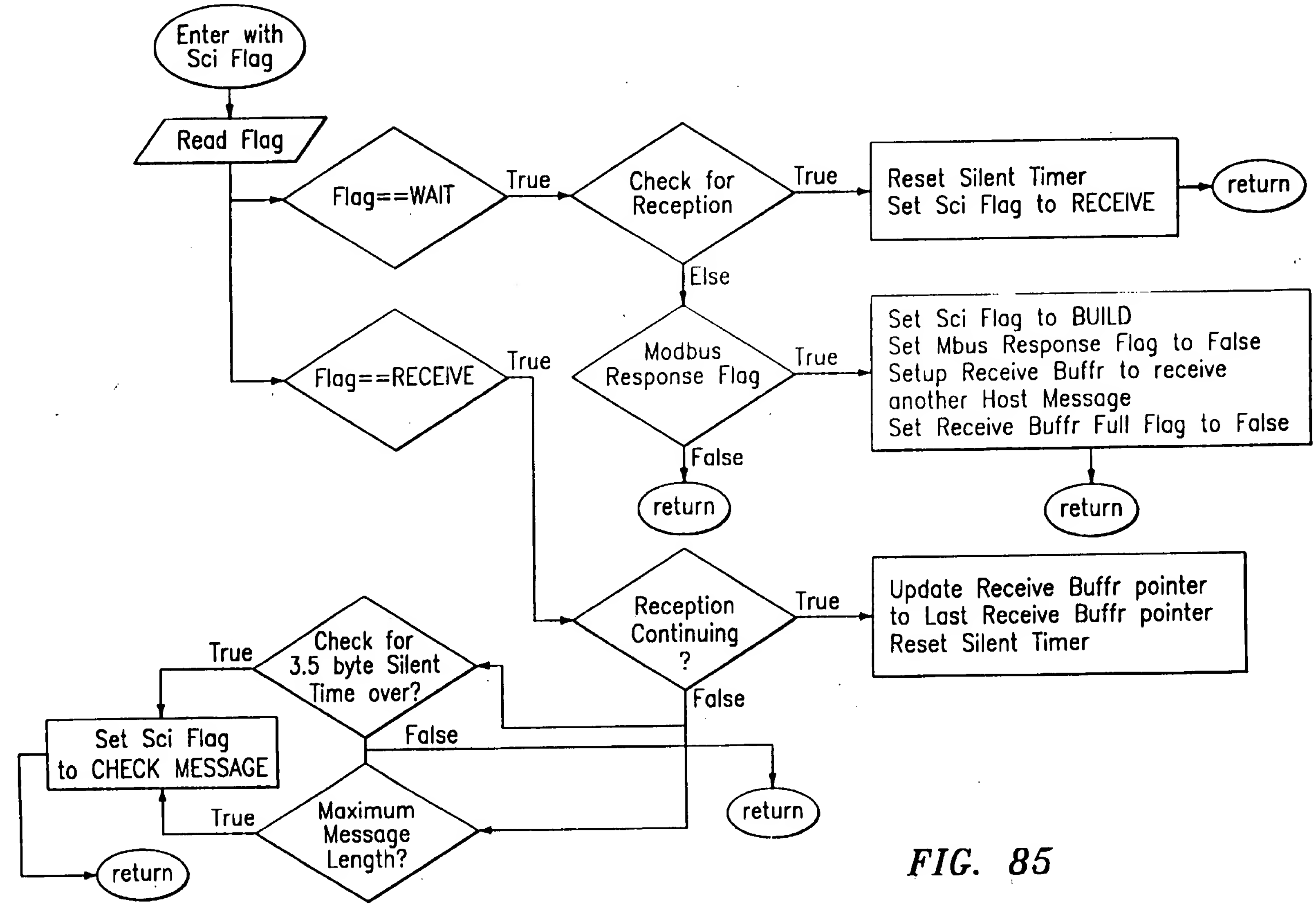

Enter with Sci Flag
Read Flag
Flag==WAIT
True
Check for Reception
True
Reset Silent Timer
Set Sci Flag to RECEIVE
return
Else
Modbus Response Flag
True
Set Sci Flag to BUILD
Set Mbus Response Flag to False
Setup Receive Buffr to receive
another Host Message
Set Receive Buffr Full Flag to False
return
False
return
Flag==RECEIVE
True
Reception Continuing ?
True
Update Receive Buffr pointer
to Last Receive Buffr pointer
Reset Silent Timer
False
Check for 3.5 byte Silent Time over?
True
False
return
Maximum Message Length?
True
Set Sci Flag to CHECK MESSAGE
return

FIG. 85

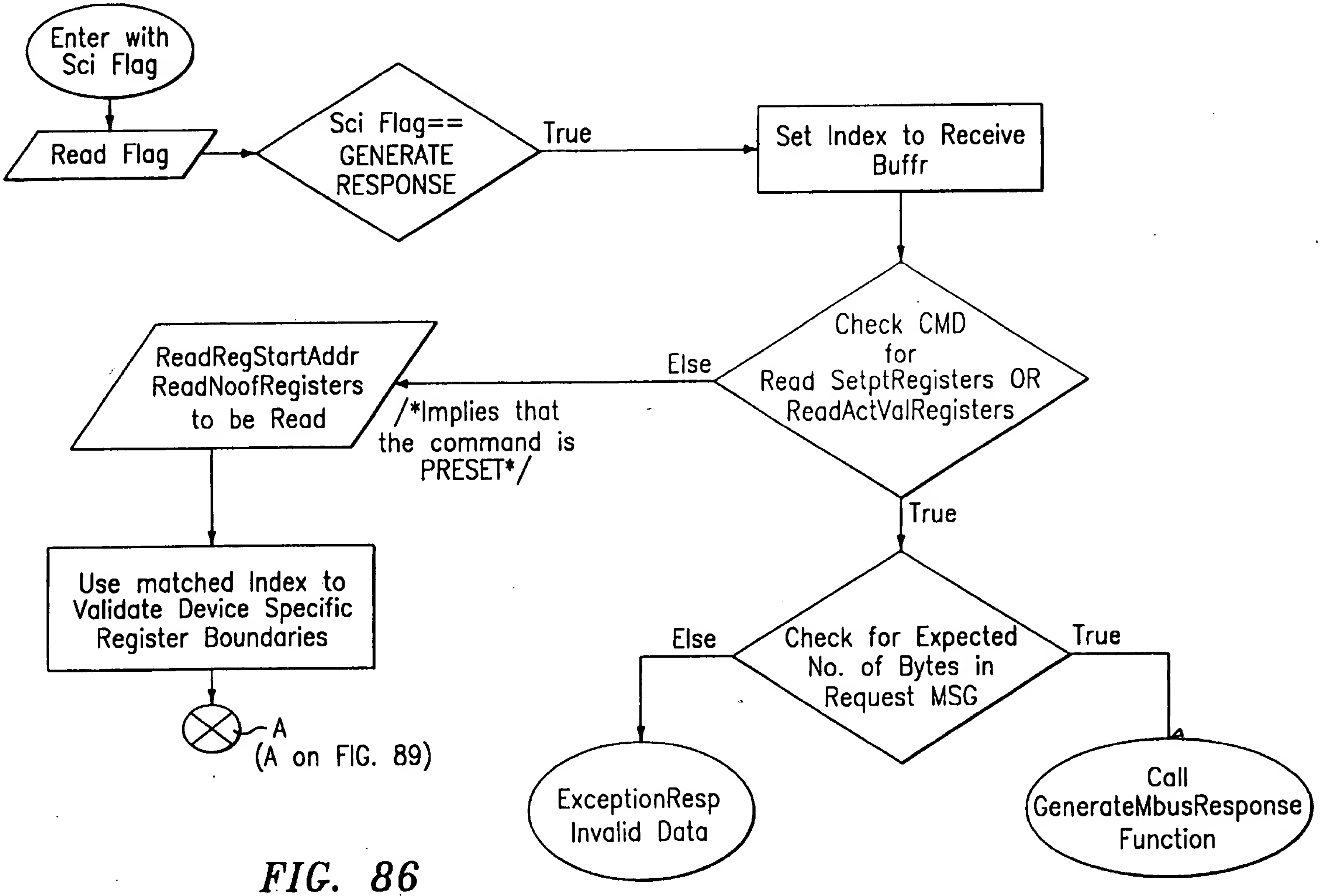

Enter with Sci Flag
Read Flag
Sci Flag== GENERATE RESPONSE
True
Set Index to Receive Buffr
Check CMD for Read SetptRegisters OR ReadActValRegisters
Else
ReadRegStartAddr ReadNoofRegisters to be Read
/*Implies that the command is PRESET*/
Use matched Index to Validate Device Specific Register Boundaries
A
(A on FIG. 89)
True
Check for Expected No. of Bytes in Request MSG
Else
True
ExceptionResp Invalid Data
Call GenerateMbusResponse Function

FIG. 86

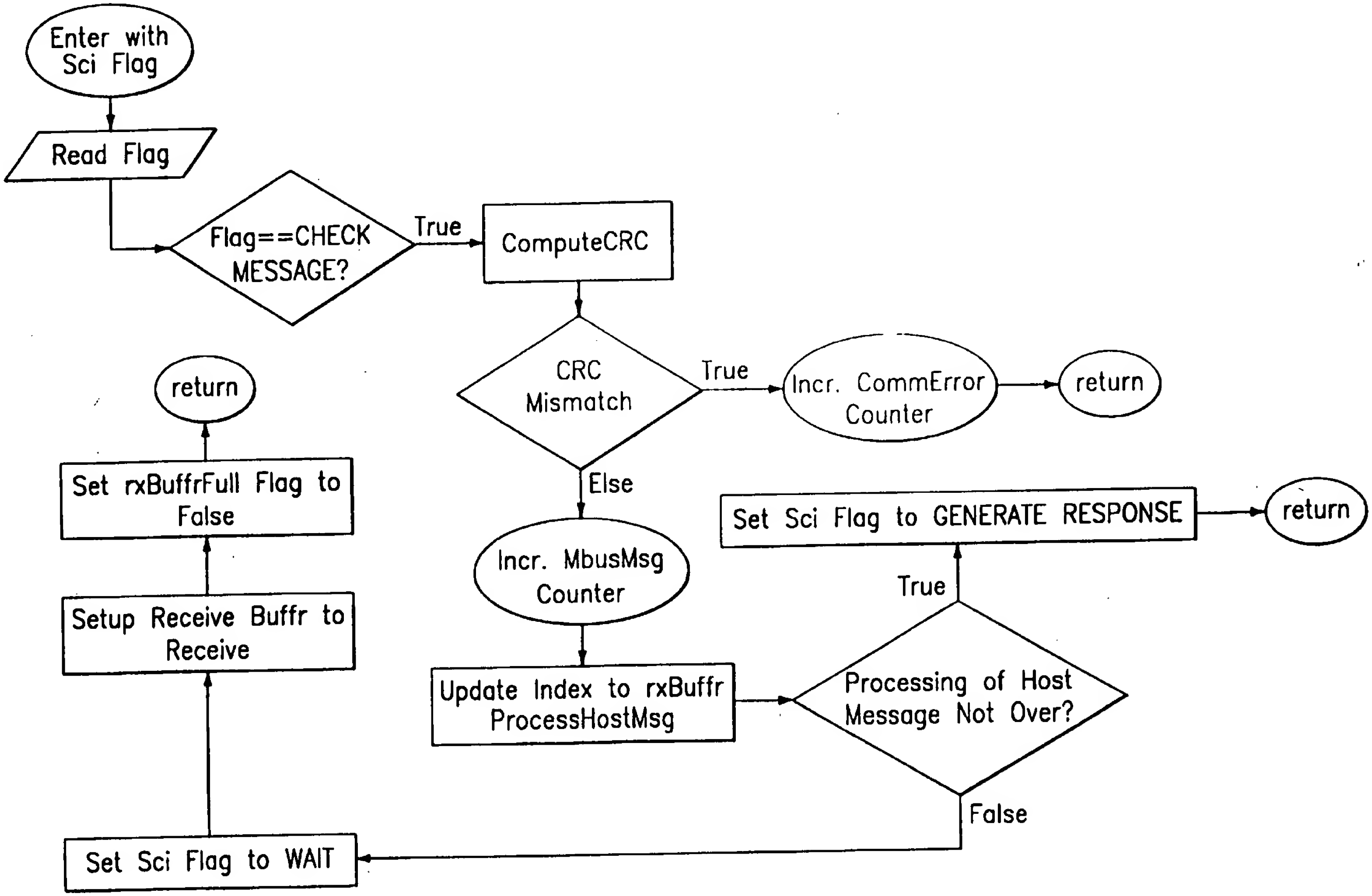

Enter with Sci Flag
Read Flag
Flag==CHECK MESSAGE?
True
ComputeCRC
CRC Mismatch
True
Incr. CommError Counter
return
Else
Incr. MbusMsg Counter
Update Index to rxBuffr ProcessHostMsg
Processing of Host Message Not Over?
True
Set Sci Flag to GENERATE RESPONSE
return
False
Set Sci Flag to WAIT
Setup Receive Buffr to Receive
Set rxBuffrFull Flag to False
return

FIG. 87

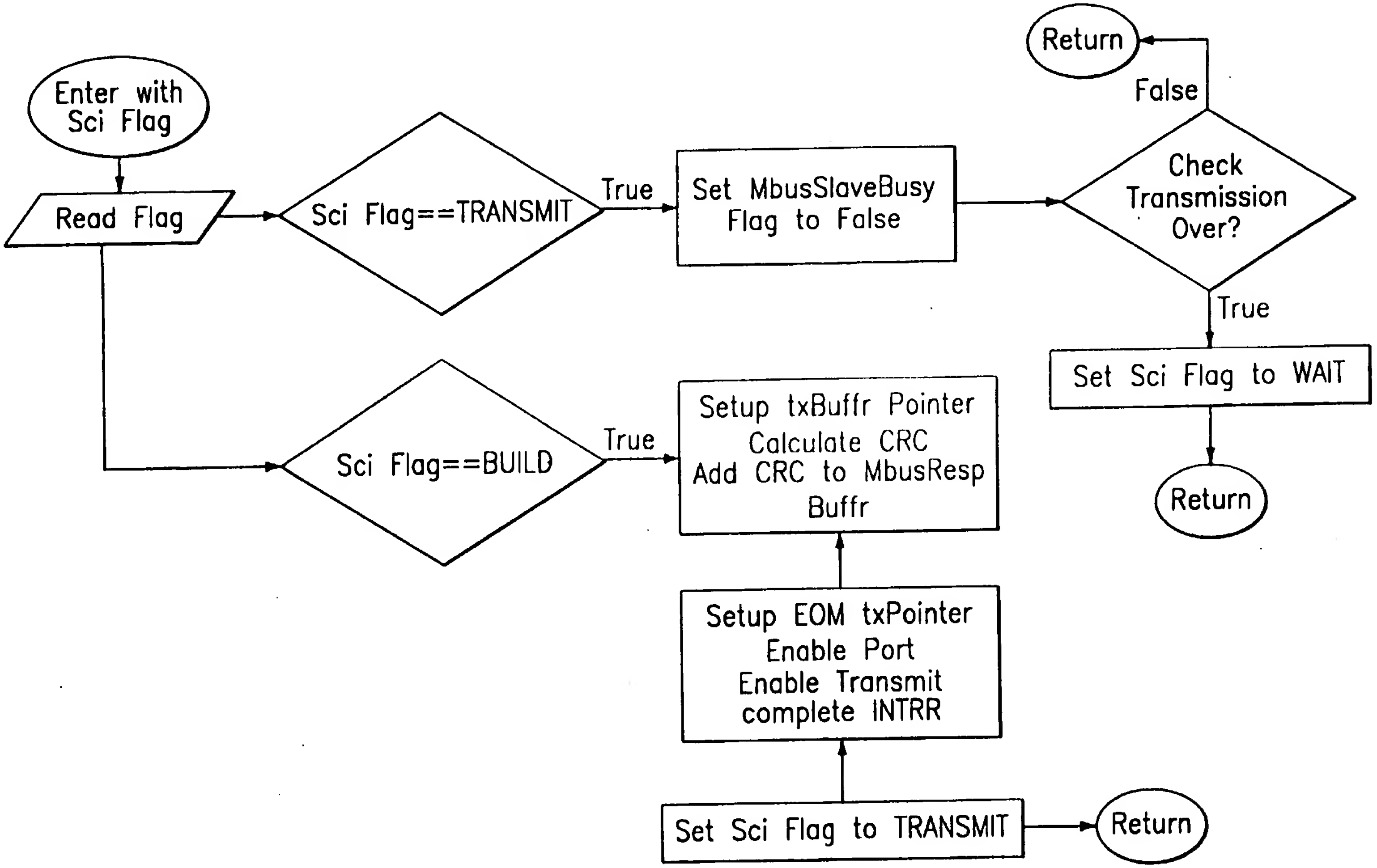

Enter with Sci Flag
Read Flag
Sci Flag==TRANSMIT
True
Set MbusSlaveBusy Flag to False
Check Transmission Over?
False
Return
True
Set Sci Flag to WAIT
Return
Sci Flag==BUILD
True
Setup txBuffr Pointer
Calculate CRC
Add CRC to MbusResp Buffr
Setup EOM txPointer
Enable Port
Enable Transmit complete INTRR
Set Sci Flag to TRANSMIT
Return

FIG. 88

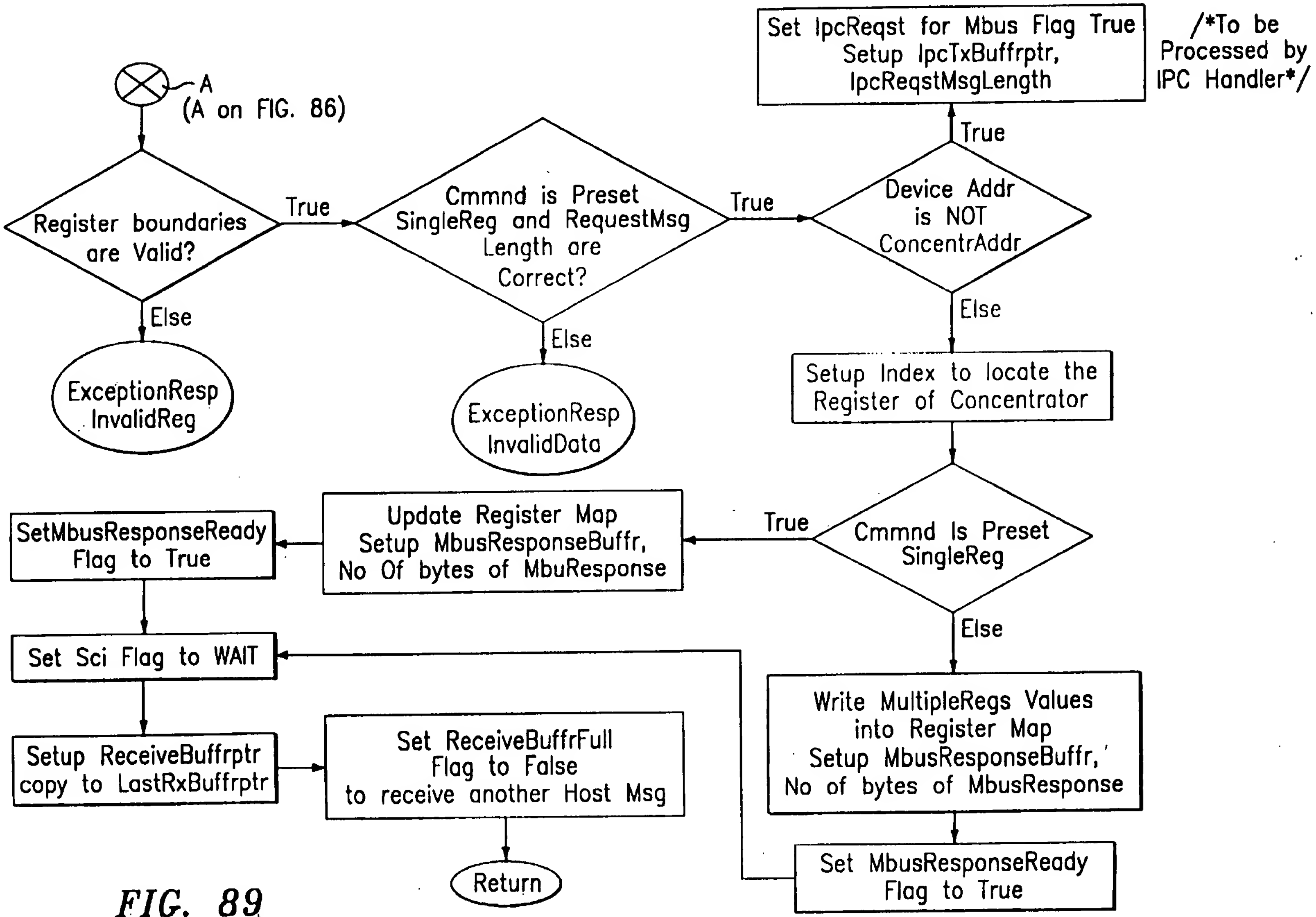
A
(A on FIG. 86)
Register boundaries are Valid?
True
Else
ExceptionResp InvalidReg
Cmmnd is Preset SingleReg and RequestMsg Length are Correct?
True
Else
ExceptionResp InvalidData
Device Addr is NOT ConcentrAddr
True
Set IpcReqst for Mbus Flag True Setup IpcTxBuffrptr, IpcReqstMsgLength
/*To be Processed by IPC Handler*/
Else
Setup Index to locate the Register of Concentrator
Cmmnd Is Preset SingleReg
True
Update Register Map Setup MbusResponseBuffr, No Of bytes of MbuResponse
SetMbusResponseReady Flag to True
Else
Write MultipleRegs Values into Register Map Setup MbusResponseBuffr, No of bytes of MbusResponse
Set MbusResponseReady Flag to True
Set Sci Flag to WAIT
Setup ReceiveBuffrptr copy to LastRxBuffrptr
Set ReceiveBuffrFull Flag to False to receive another Host Msg
Return

FIG. 89

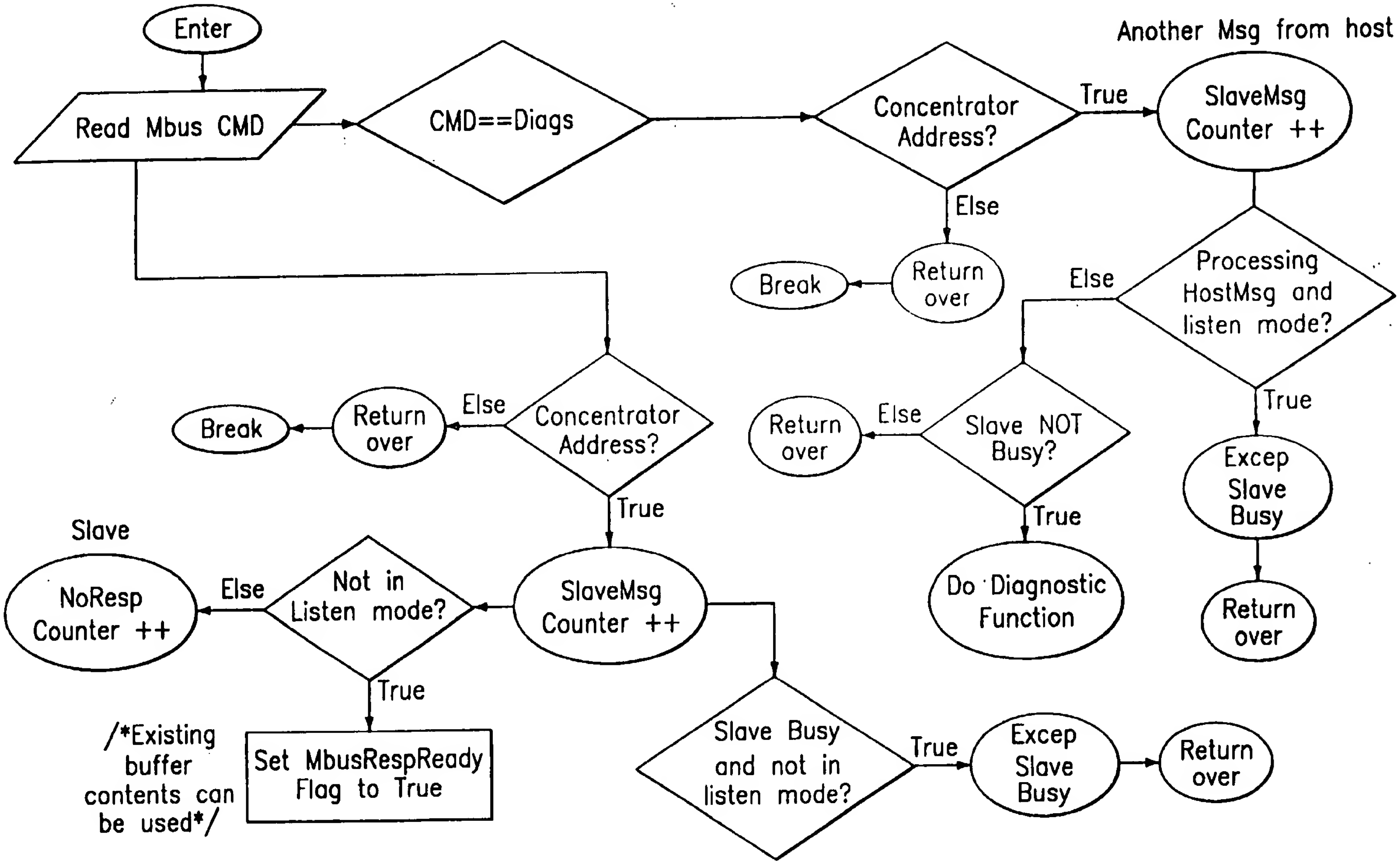

Enter
Read Mbus CMD
CMD==Diags
Concentrator Address?
True
Another Msg from host
SlaveMsg Counter ++
Else
Return over
Break
Processing HostMsg and listen mode?
Else
True
Excep Slave Busy
Return over
Slave NOT Busy?
Else
Return over
True
Do Diagnostic Function
Concentrator Address?
Else
Return over
Break
True
SlaveMsg Counter ++
Not in Listen mode?
Else
Slave
NoResp Counter ++
True
Set MbusRespReady Flag to True
/*Existing buffer contents can be used*/
Slave Busy and not in listen mode?
True
Excep Slave Busy
Return over

FIG. 90

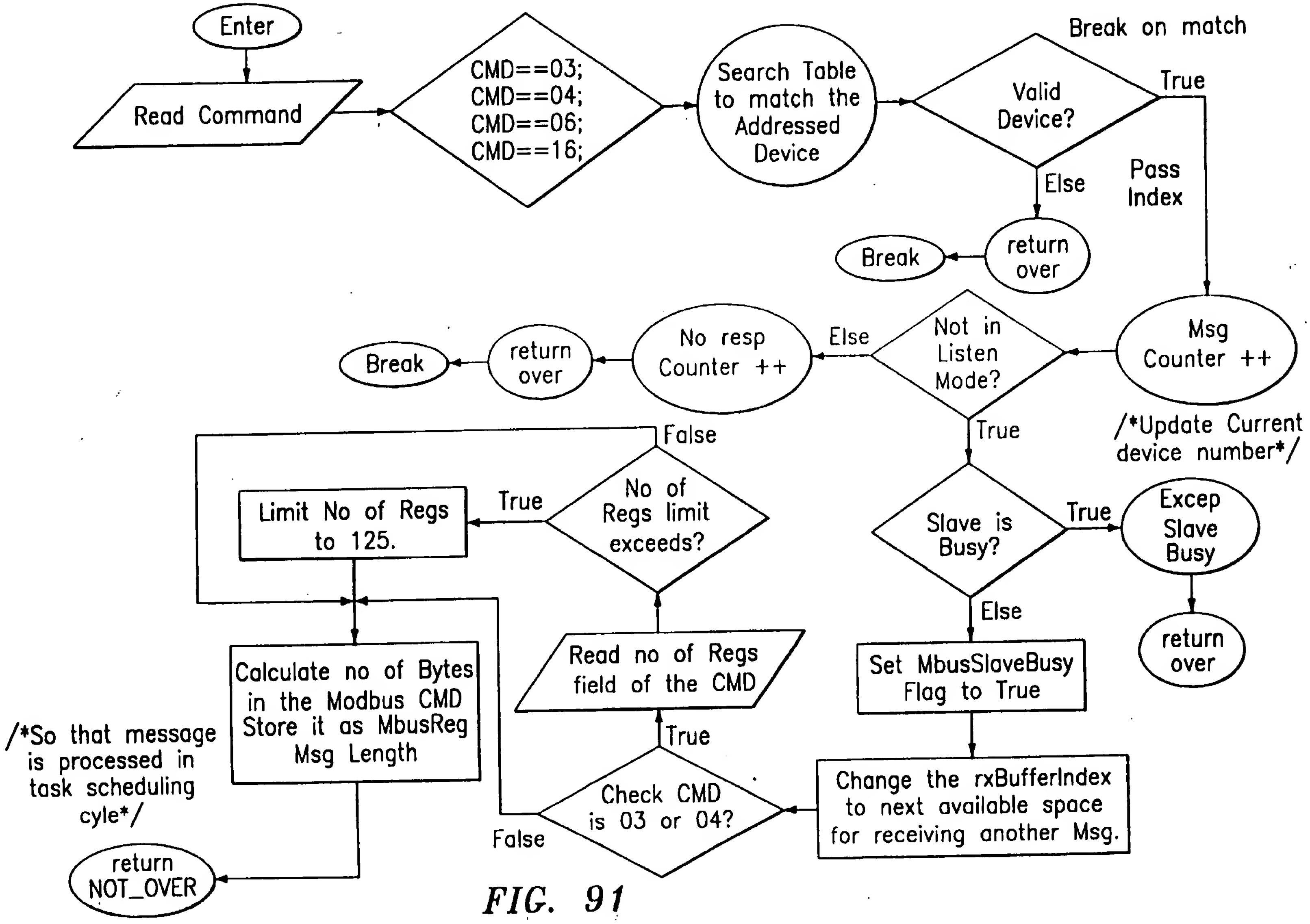

Enter
Read Command
CMD==03;
CMD==04;
CMD==06;
CMD==16;
Search Table to match the Addressed Device
Break on match
Valid Device?
True
Else
Pass Index
return over
Break
Msg Counter ++
/*Update Current device number*/
Not in Listen Mode?
Else
No resp Counter ++
return over
Break
True
Slave is Busy?
True
Excep Slave Busy
return over
Else
Set MbusSlaveBusy Flag to True
Change the rxBufferIndex to next available space for receiving another Msg.
Check CMD is 03 or 04?
True
Read no of Regs field of the CMD
No of Regs limit exceeds?
True
Limit No of Regs to 125.
False
False
Calculate no of Bytes in the Modbus CMD Store it as MbusReg Msg Length
/*So that message is processed in task scheduling cyle*/
return NOT_OVER

FIG. 91

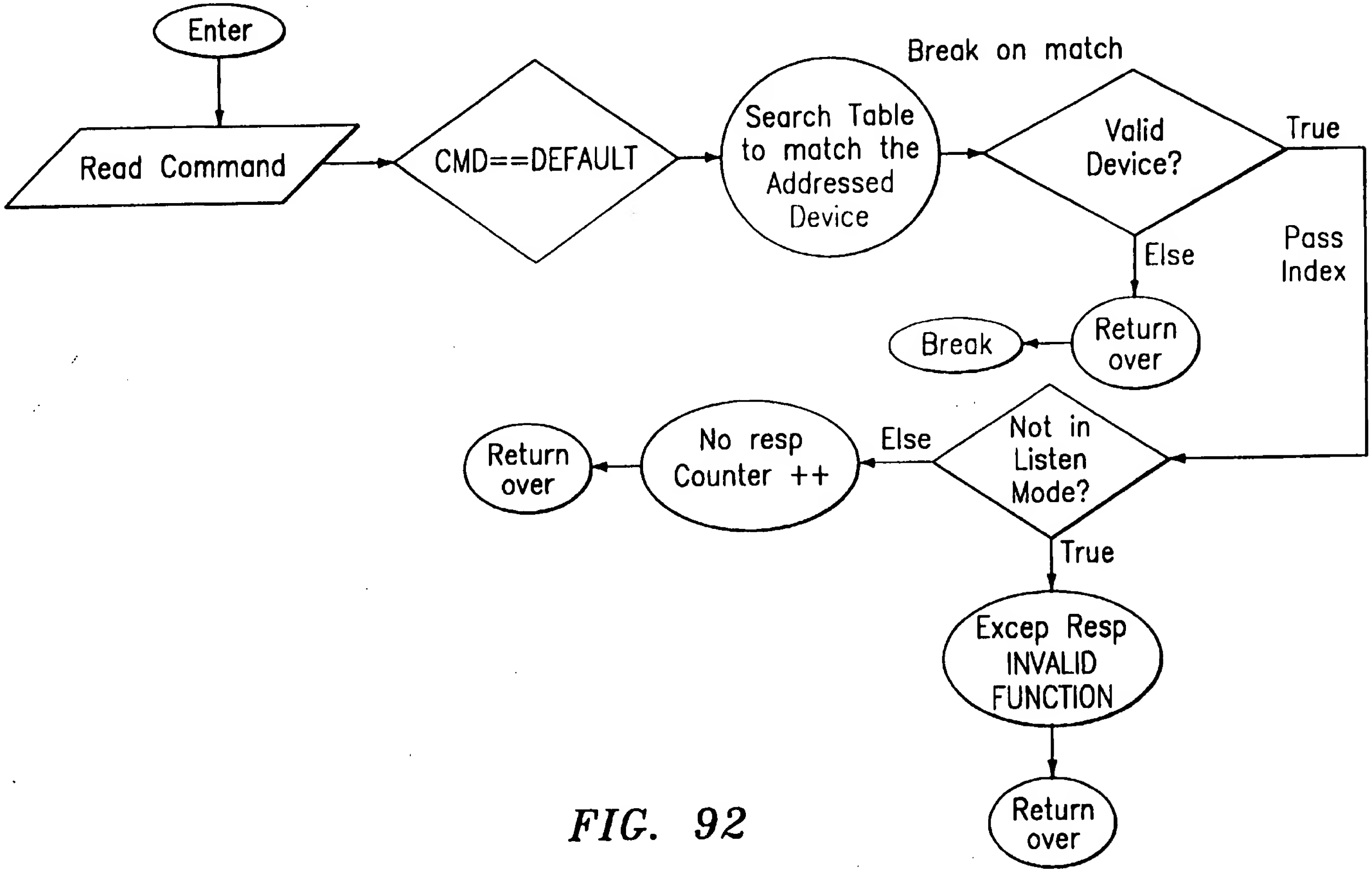

Enter
Read Command
CMD==DEFAULT
Search Table to match the Addressed Device
Break on match
Valid Device?
True
Pass Index
Else
Return over
Break
Not in Listen Mode?
Else
No resp Counter ++
Return over
True
Excep Resp INVALID FUNCTION
Return over

FIG. 92

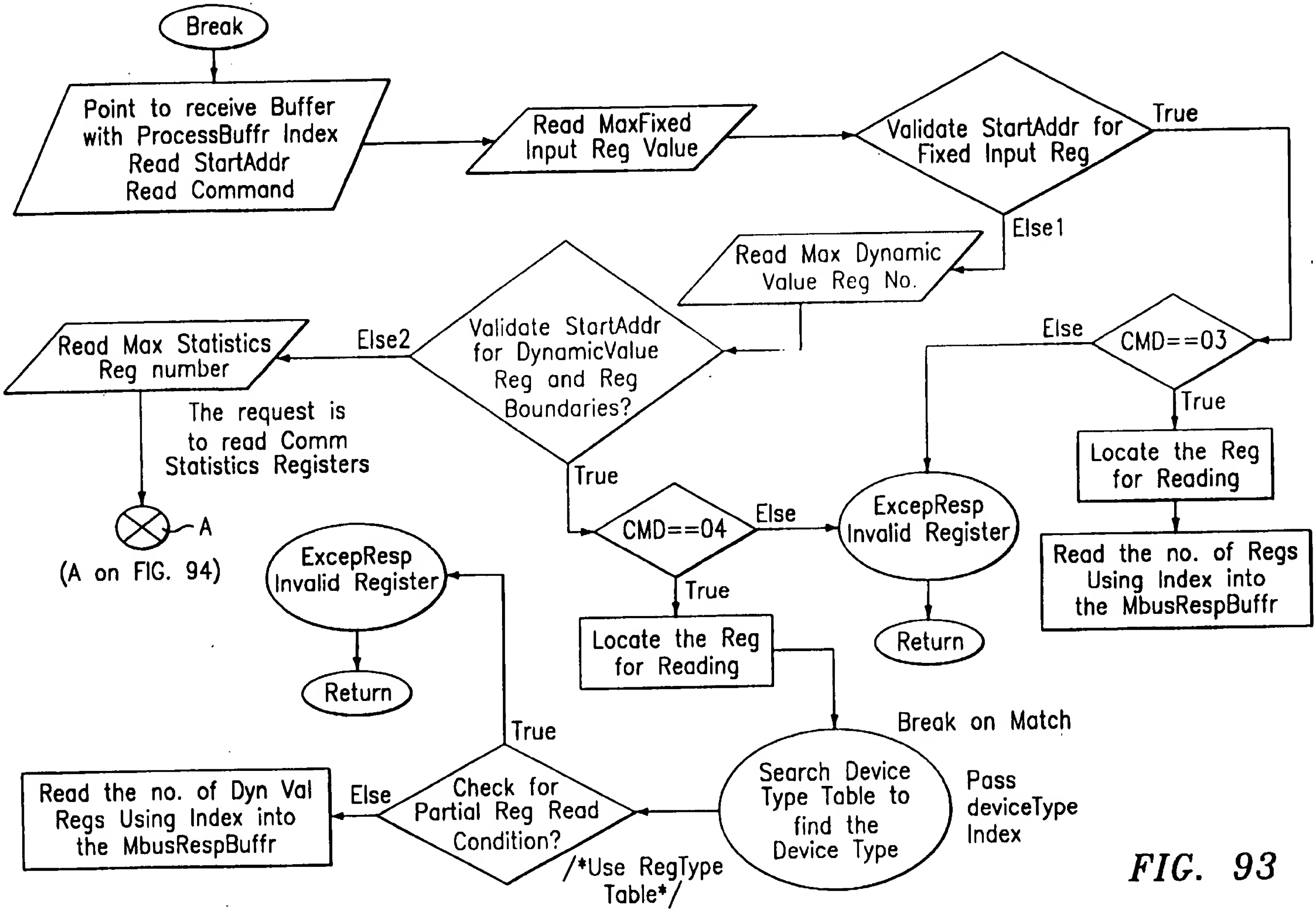

Break
Point to receive Buffer with ProcessBuffr Index Read StartAddr Read Command
Read MaxFixed Input Reg Value
Validate StartAddr for Fixed Input Reg
True
Else1
Read Max Dynamic Value Reg No.
CMD==03
Else
True
Locate the Reg for Reading
Read the no. of Regs Using Index into the MbusRespBuffr
Validate StartAddr for DynamicValue Reg and Reg Boundaries?
Else2
Read Max Statistics Reg number
The request is to read Comm Statistics Registers
A
(A on FIG. 94)
True
CMD==04
Else
ExcepResp Invalid Register
Return
True
Locate the Reg for Reading
Break on Match
Search Device Type Table to find the Device Type
Pass deviceType Index
Check for Partial Reg Read Condition?
/*Use RegType Table*/
True
ExcepResp Invalid Register
Return
Else
Read the no. of Dyn Val Regs Using Index into the MbusRespBuffr

FIG. 93

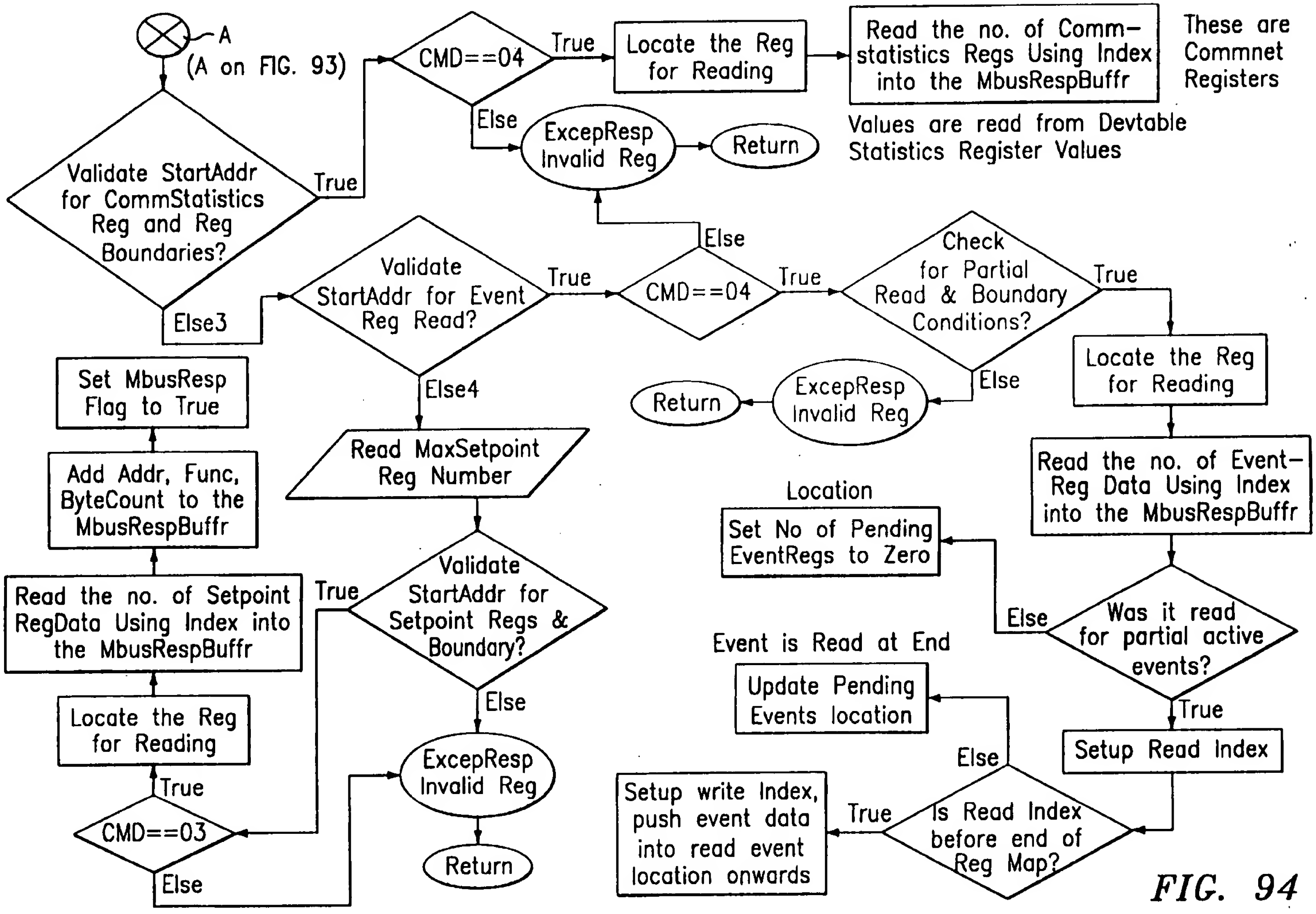

A
(A on FIG. 93)
Validate StartAddr for CommStatistics Reg and Reg Boundaries?
True
Else3
CMD==04
True
Else
Locate the Reg for Reading
Read the no. of Comm-statistics Regs Using Index into the MbusRespBuffr
These are Commnet Registers
Values are read from Devtable Statistics Register Values
ExcepResp Invalid Reg
Return
Validate StartAddr for Event Reg Read?
True
Else4
CMD==04
True
Else
Check for Partial Read & Boundary Conditions?
True
Else
ExcepResp Invalid Reg
Return
Locate the Reg for Reading
Read the no. of Event-Reg Data Using Index into the MbusRespBuffr
Was it read for partial active events?
Else
True
Location
Set No of Pending EventRegs to Zero
Setup Read Index
Is Read Index before end of Reg Map?
Else
True
Event is Read at End
Update Pending Events location
Setup write Index, push event data into read event location onwards
Read MaxSetpoint Reg Number
Validate StartAddr for Setpoint Regs & Boundary?
True
Else
ExcepResp Invalid Reg
Return
CMD==03
True
Else
Locate the Reg for Reading
Read the no. of Setpoint RegData Using Index into the MbusRespBuffr
Add Addr, Func, ByteCount to the MbusRespBuffr
Set MbusResp Flag to True

FIG. 94

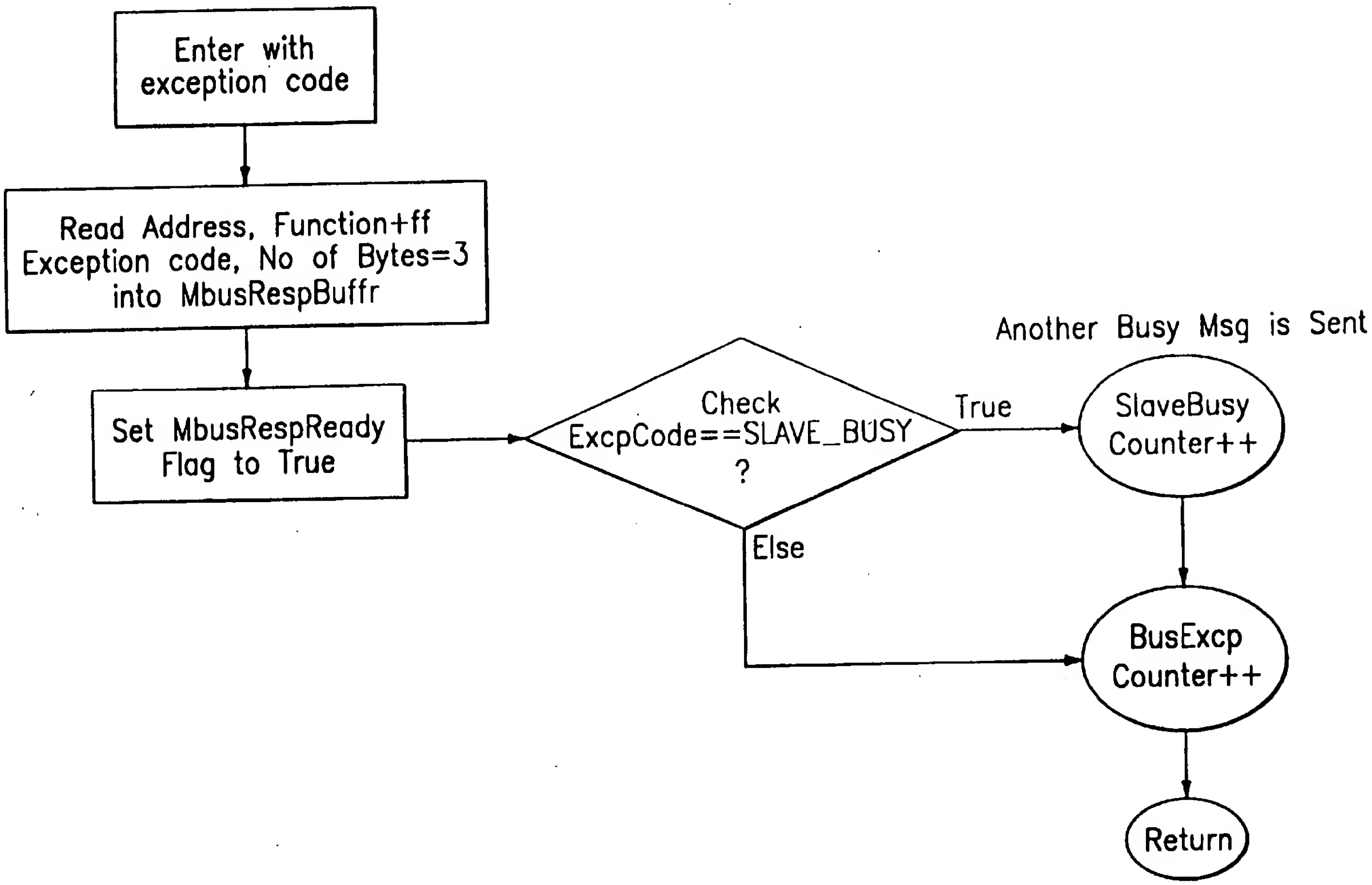

Enter with exception code
Read Address, Function+ff Exception code, No of Bytes=3 into MbusRespBuffr
Set MbusRespReady Flag to True
Check ExcpCode==SLAVE_BUSY ?
True
Else
Another Busy Msg is Sent
SlaveBusy Counter++
BusExcp Counter++
Return

FIG. 95

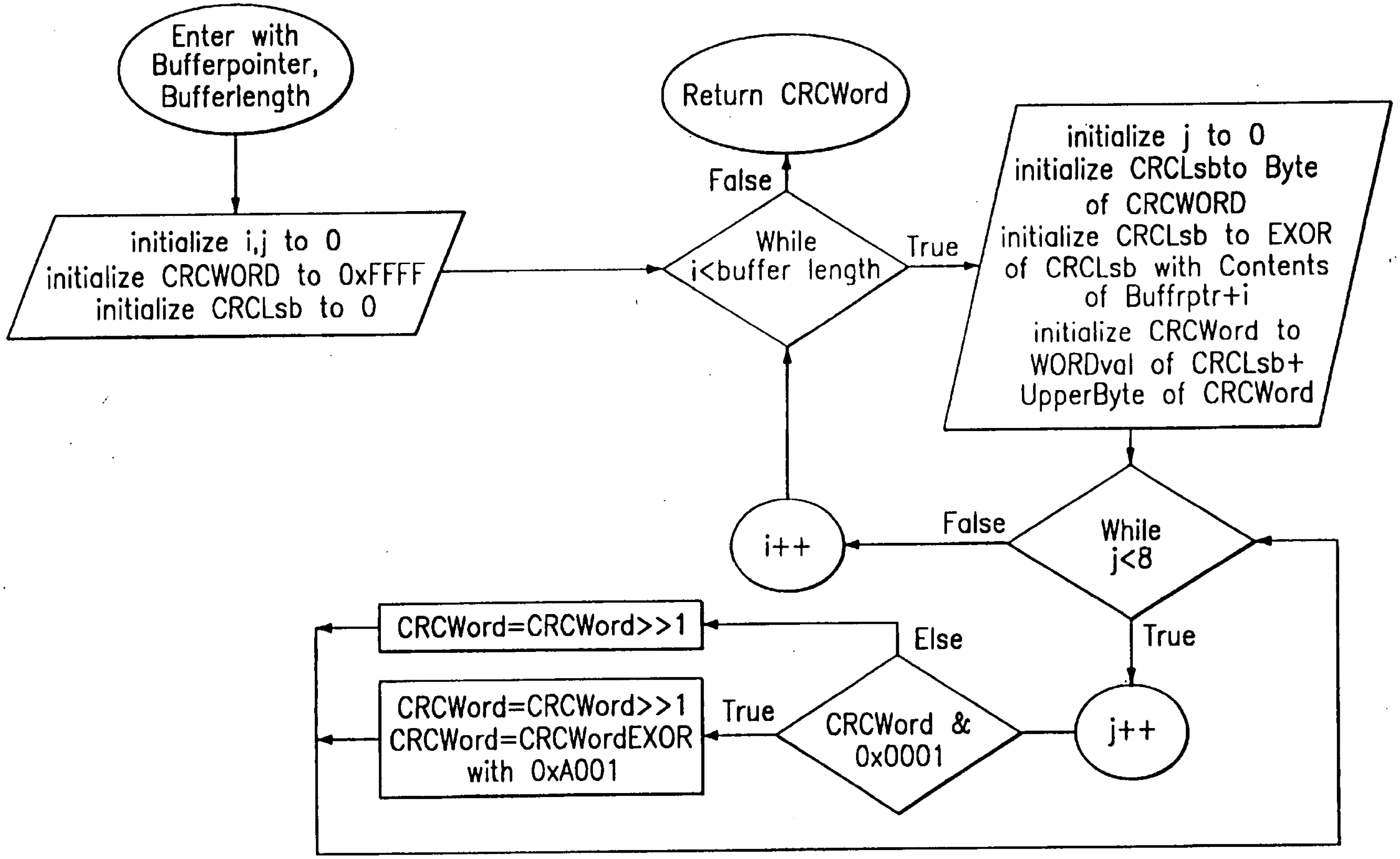

Enter with
Bufferpointer,
Bufferlength
initialize i,j to 0
initialize CRCWORD to 0xFFFF
initialize CRCLsb to 0
Return CRCWord
False
While
i<buffer length
True
initialize j to 0
initialize CRCLsbto Byte
of CRCWORD
initialize CRCLsb to EXOR
of CRCLsb with Contents
of Buffrptr+i
initialize CRCWord to
WORDval of CRCLsb+
UpperByte of CRCWord
i++
False
While
j<8
True
j++
CRCWord &
0x0001
Else
True
CRCWord=CRCWord>>1
CRCWord=CRCWord>>1
CRCWord=CRCWordEXOR
with 0xA001

FIG. 96

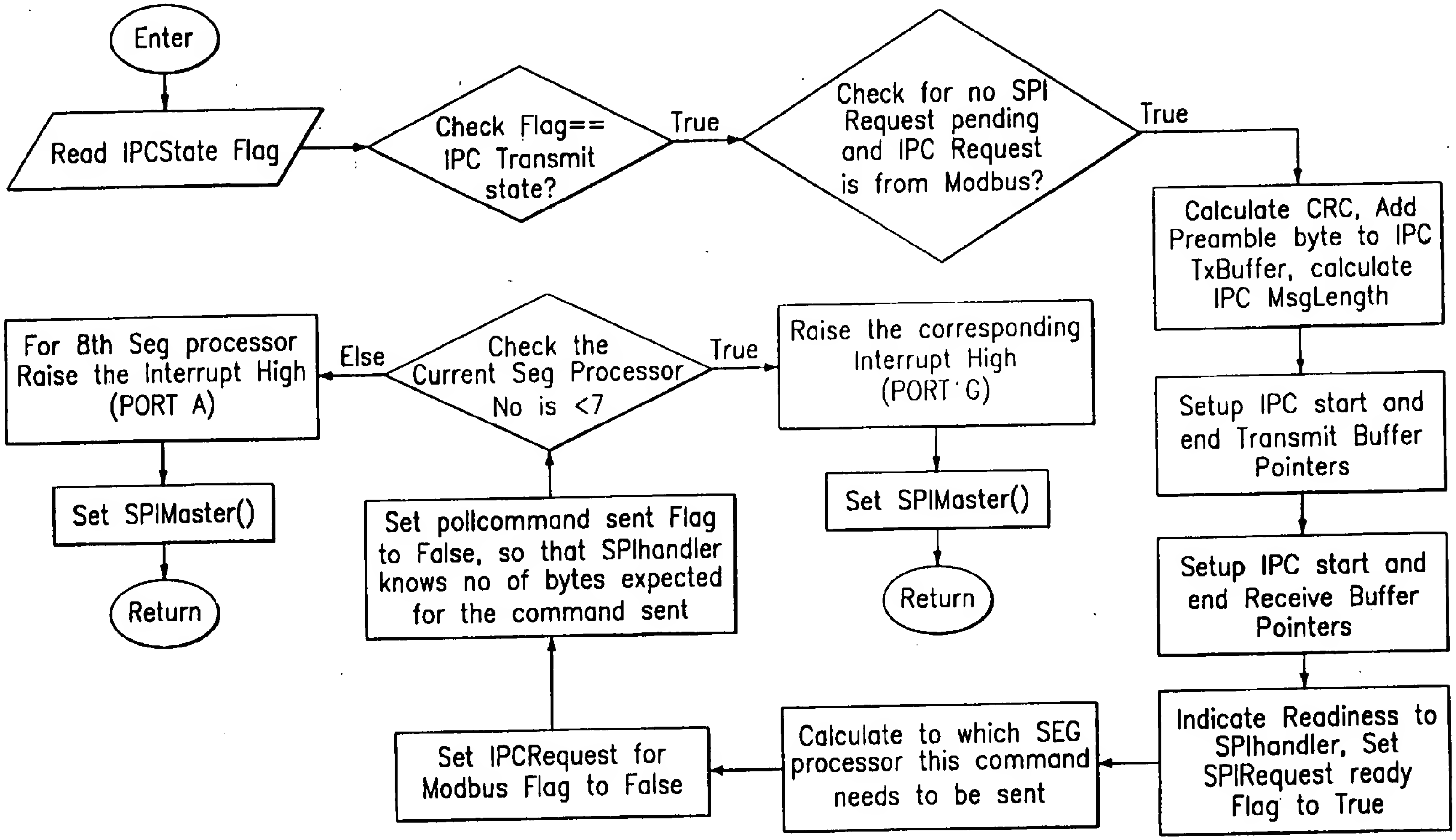

Enter
Read IPCState Flag
Check Flag== IPC Transmit state?
True
Check for no SPI Request pending and IPC Request is from Modbus?
True
Calculate CRC, Add Preamble byte to IPC TxBuffer, calculate IPC MsgLength
Setup IPC start and end Transmit Buffer Pointers
Setup IPC start and end Receive Buffer Pointers
Indicate Readiness to SPIhandler, Set SPIRequest ready Flag to True
Calculate to which SEG processor this command needs to be sent
Set IPCRequest for Modbus Flag to False
Set pollcommand sent Flag to False, so that SPIhandler knows no of bytes expected for the command sent
Check the Current Seg Processor No is <7
True
Raise the corresponding Interrupt High (PORT G)
Set SPIMaster()
Return
Else
For 8th Seg processor Raise the Interrupt High (PORT A)
Set SPIMaster()
Return

FIG. 97

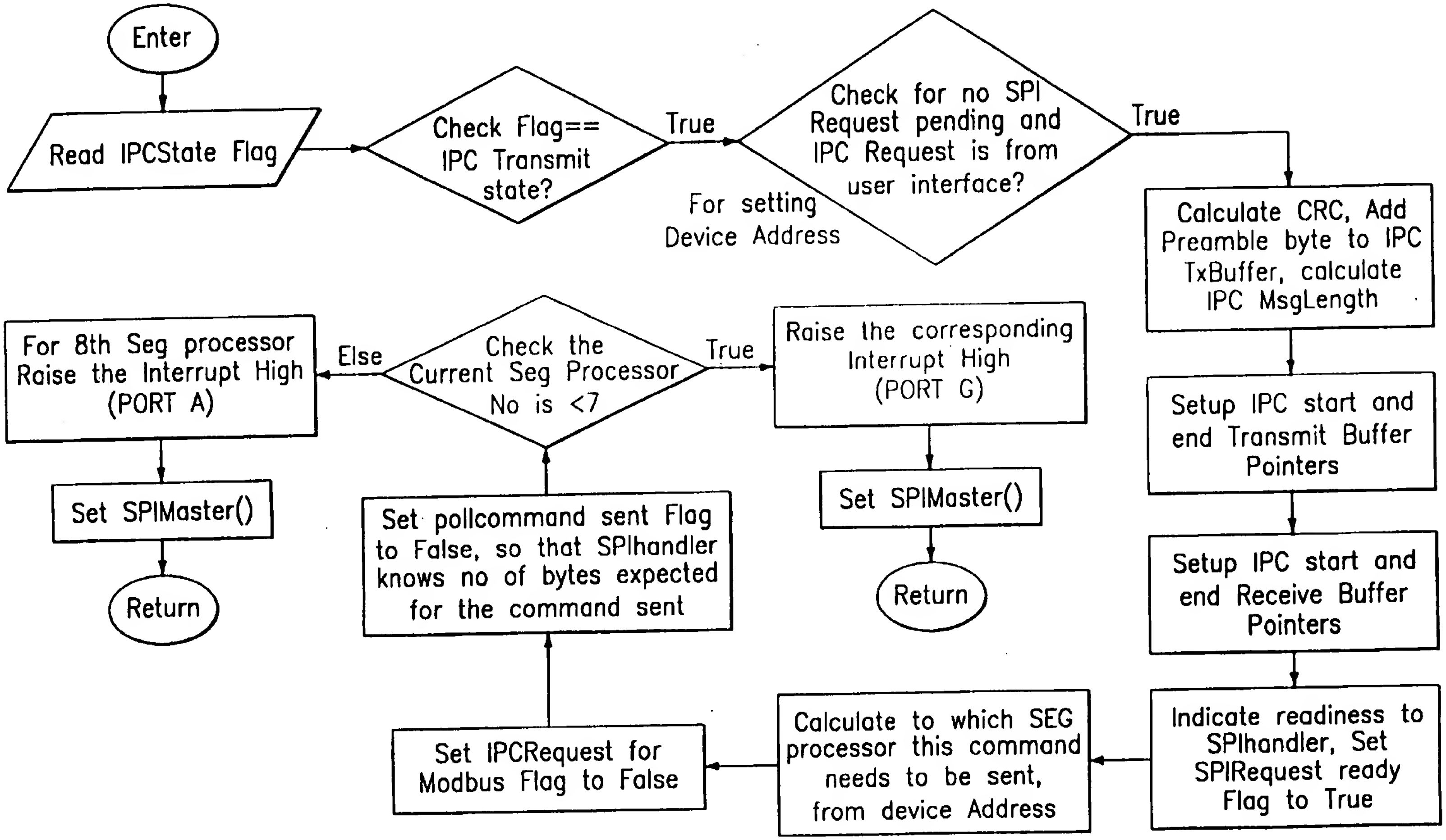

Enter
Read IPCState Flag
Check Flag== IPC Transmit state?
True
Check for no SPI Request pending and IPC Request is from user interface?
For setting Device Address
True
Calculate CRC, Add Preamble byte to IPC TxBuffer, calculate IPC MsgLength
Setup IPC start and end Transmit Buffer Pointers
Setup IPC start and end Receive Buffer Pointers
Indicate readiness to SPIhandler, Set SPIRequest ready Flag to True
Calculate to which SEG processor this command needs to be sent, from device Address
Set IPCRequest for Modbus Flag to False
Set pollcommand sent Flag to False, so that SPIhandler knows no of bytes expected for the command sent
Check the Current Seg Processor No is <7
True
Raise the corresponding Interrupt High (PORT G)
Set SPIMaster()
Return
Else
For 8th Seg processor Raise the Interrupt High (PORT A)
Set SPIMaster()
Return

FIG. 98

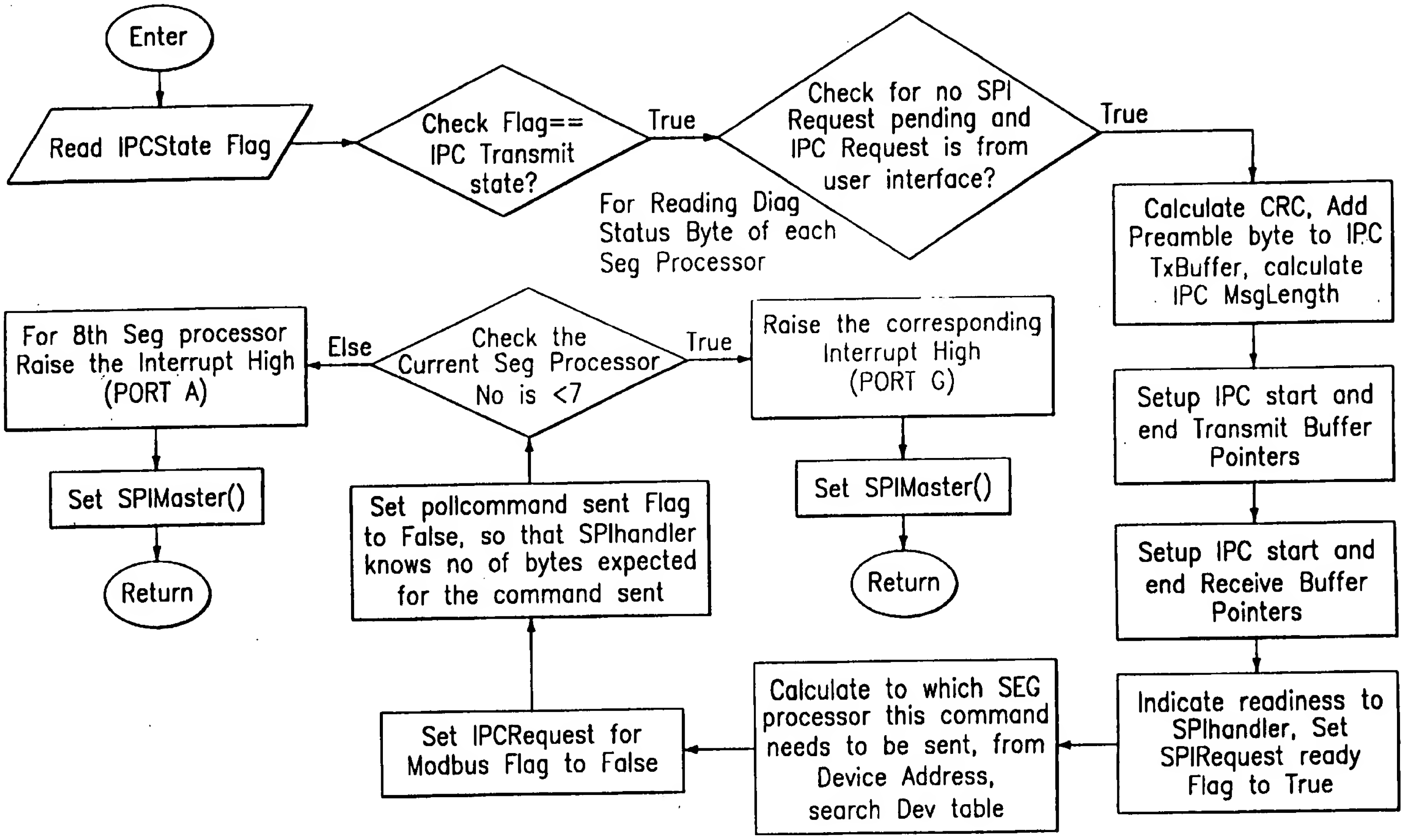

Enter
Read IPCState Flag
Check Flag== IPC Transmit state?
True
Check for no SPI Request pending and IPC Request is from user interface?
True
For Reading Diag Status Byte of each Seg Processor
Calculate CRC, Add Preamble byte to IPC TxBuffer, calculate IPC MsgLength
Setup IPC start and end Transmit Buffer Pointers
Setup IPC start and end Receive Buffer Pointers
Indicate readiness to SPIhandler, Set SPIRequest ready Flag to True
Calculate to which SEG processor this command needs to be sent, from Device Address, search Dev table
Set IPCRequest for Modbus Flag to False
Set pollcommand sent Flag to False, so that SPIhandler knows no of bytes expected for the command sent
Check the Current Seg Processor No is <7
True
Raise the corresponding Interrupt High (PORT G)
Set SPIMaster()
Return
Else
For 8th Seg processor Raise the Interrupt High (PORT A)
Set SPIMaster()
Return

FIG. 99

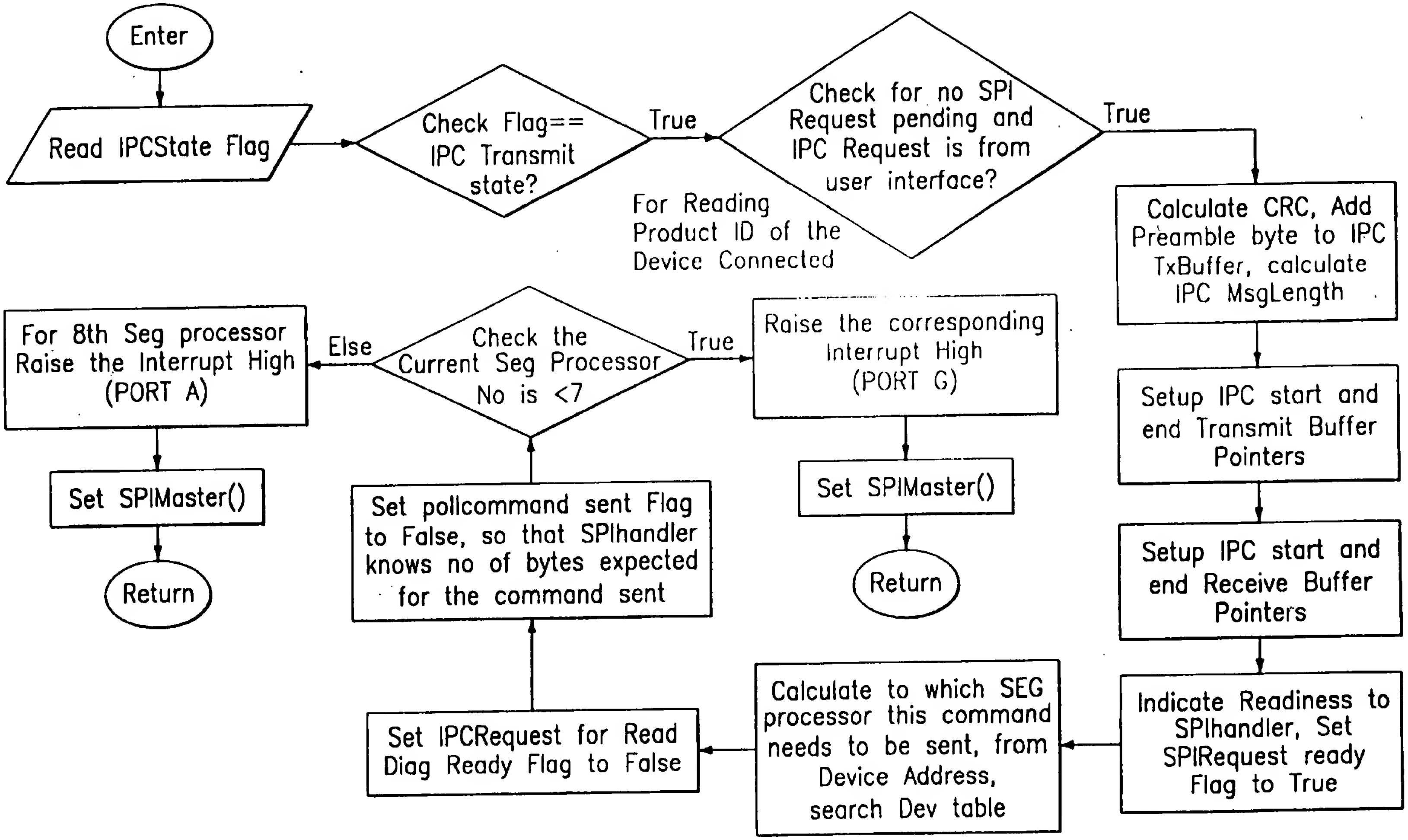

Enter
Read IPCState Flag
Check Flag== IPC Transmit state?
True
Check for no SPI Request pending and IPC Request is from user interface?
For Reading Product ID of the Device Connected
True
Calculate CRC, Add Preamble byte to IPC TxBuffer, calculate IPC MsgLength
Setup IPC start and end Transmit Buffer Pointers
Setup IPC start and end Receive Buffer Pointers
Indicate Readiness to SPIhandler, Set SPIRequest ready Flag to True
Calculate to which SEG processor this command needs to be sent, from Device Address, search Dev table
Set IPCRequest for Read Diag Ready Flag to False
Set pollcommand sent Flag to False, so that SPIhandler knows no of bytes expected for the command sent
Check the Current Seg Processor No is <7
True
Raise the corresponding Interrupt High (PORT G)
Set SPIMaster()
Return
Else
For 8th Seg processor Raise the Interrupt High (PORT A)
Set SPIMaster()
Return

FIG. 100

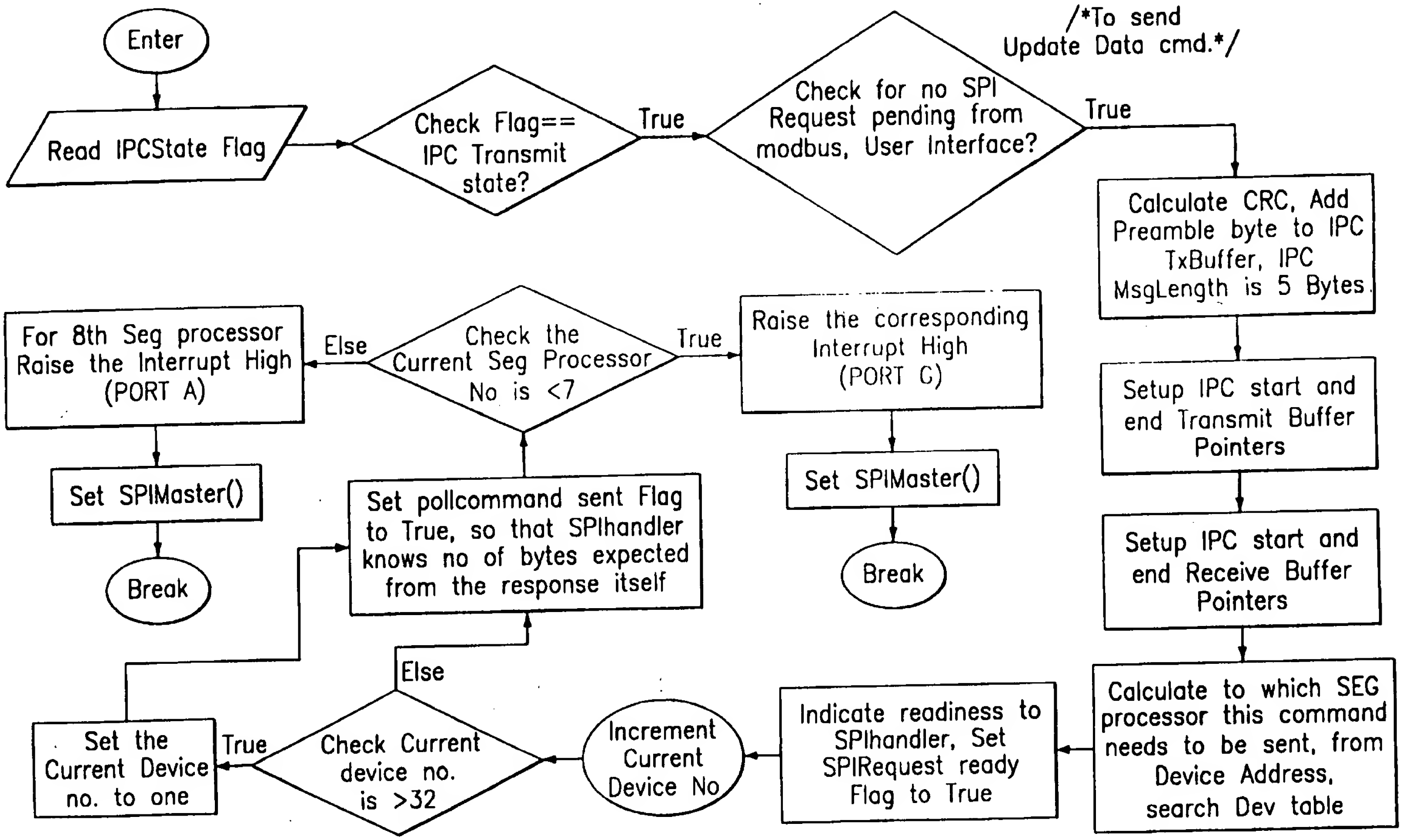

Enter
Read IPCState Flag
Check Flag== IPC Transmit state?
True
Check for no SPI Request pending from modbus, User Interface?
/*To send Update Data cmd.*/
True
Calculate CRC, Add Preamble byte to IPC TxBuffer, IPC MsgLength is 5 Bytes
Setup IPC start and end Transmit Buffer Pointers
Setup IPC start and end Receive Buffer Pointers
Calculate to which SEG processor this command needs to be sent, from Device Address, search Dev table
Indicate readiness to SPIhandler, Set SPIRequest ready Flag to True
Increment Current Device No
Check Current device no. is >32
True
Set the Current Device no. to one
Else
Set pollcommand sent Flag to True, so that SPIhandler knows no of bytes expected from the response itself
Check the Current Seg Processor No is <7
True
Raise the corresponding Interrupt High (PORT C)
Set SPIMaster()
Break
Else
For 8th Seg processor Raise the Interrupt High (PORT A)
Set SPIMaster()
Break

FIG. 101

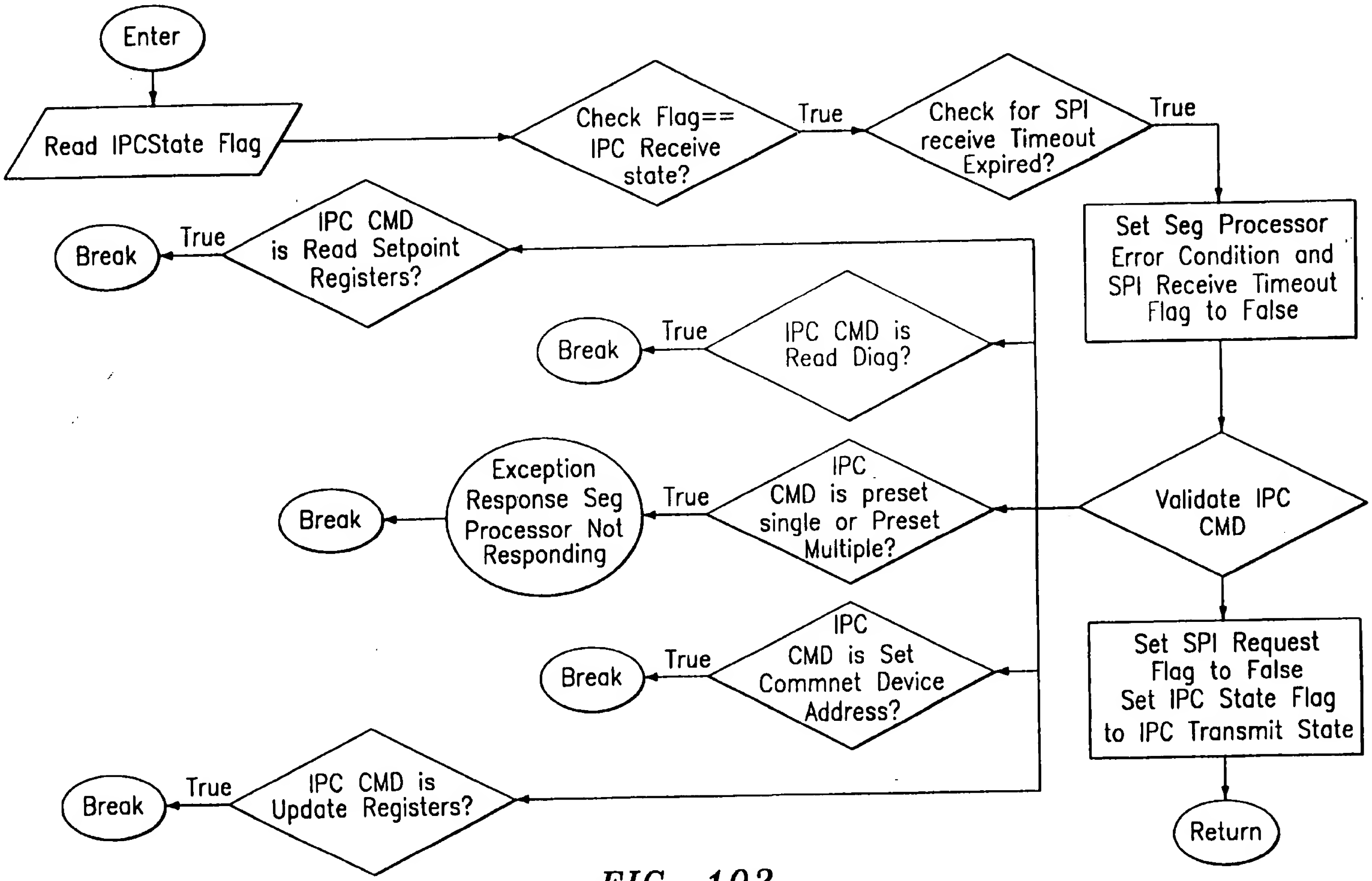

Enter
Read IPCState Flag
Check Flag== IPC Receive state?
True
Check for SPI receive Timeout Expired?
True
Set Seg Processor Error Condition and SPI Receive Timeout Flag to False
Validate IPC CMD
Set SPI Request Flag to False Set IPC State Flag to IPC Transmit State
Return
IPC CMD is Read Setpoint Registers?
True
Break
IPC CMD is Read Diag?
True
Break
IPC CMD is preset single or Preset Multiple?
True
Exception Response Seg Processor Not Responding
Break
IPC CMD is Set Commnet Device Address?
True
Break
IPC CMD is Update Registers?
True
Break

FIG. 102

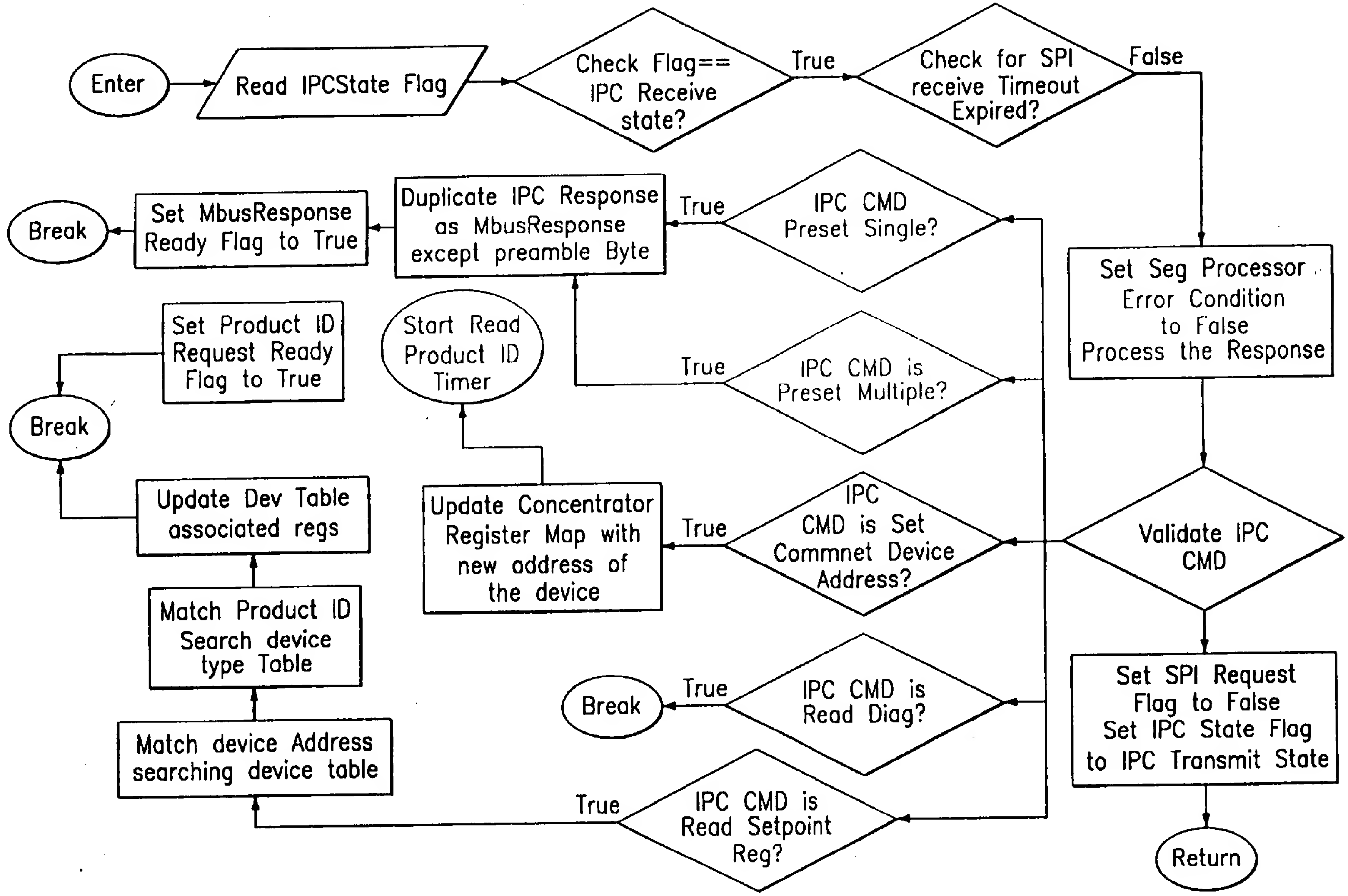

Enter
Read IPCState Flag
Check Flag== IPC Receive state?
True
Check for SPI receive Timeout Expired?
False
Set Seg Processor Error Condition to False Process the Response
Validate IPC CMD
Set SPI Request Flag to False Set IPC State Flag to IPC Transmit State
Return
IPC CMD Preset Single?
True
Duplicate IPC Response as MbusResponse except preamble Byte
Set MbusResponse Ready Flag to True
Break
IPC CMD is Preset Multiple?
True
IPC CMD is Set Commnet Device Address?
True
Update Concentrator Register Map with new address of the device
Start Read Product ID Timer
Set Product ID Request Ready Flag to True
Break
IPC CMD is Read Diag?
True
Break
IPC CMD is Read Setpoint Reg?
True
Match device Address searching device table
Match Product ID Search device type Table
Update Dev Table associated regs

FIG. 103

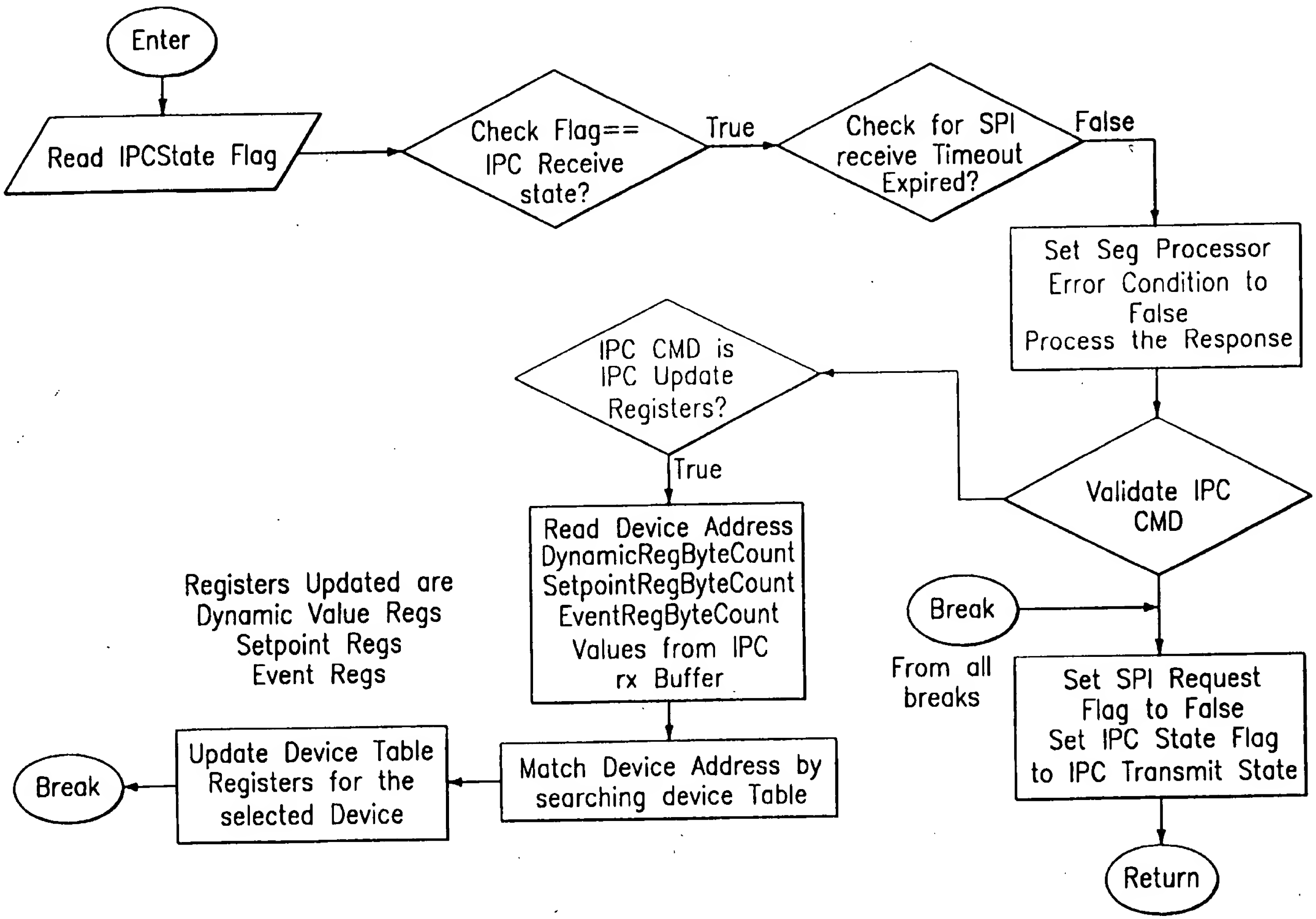
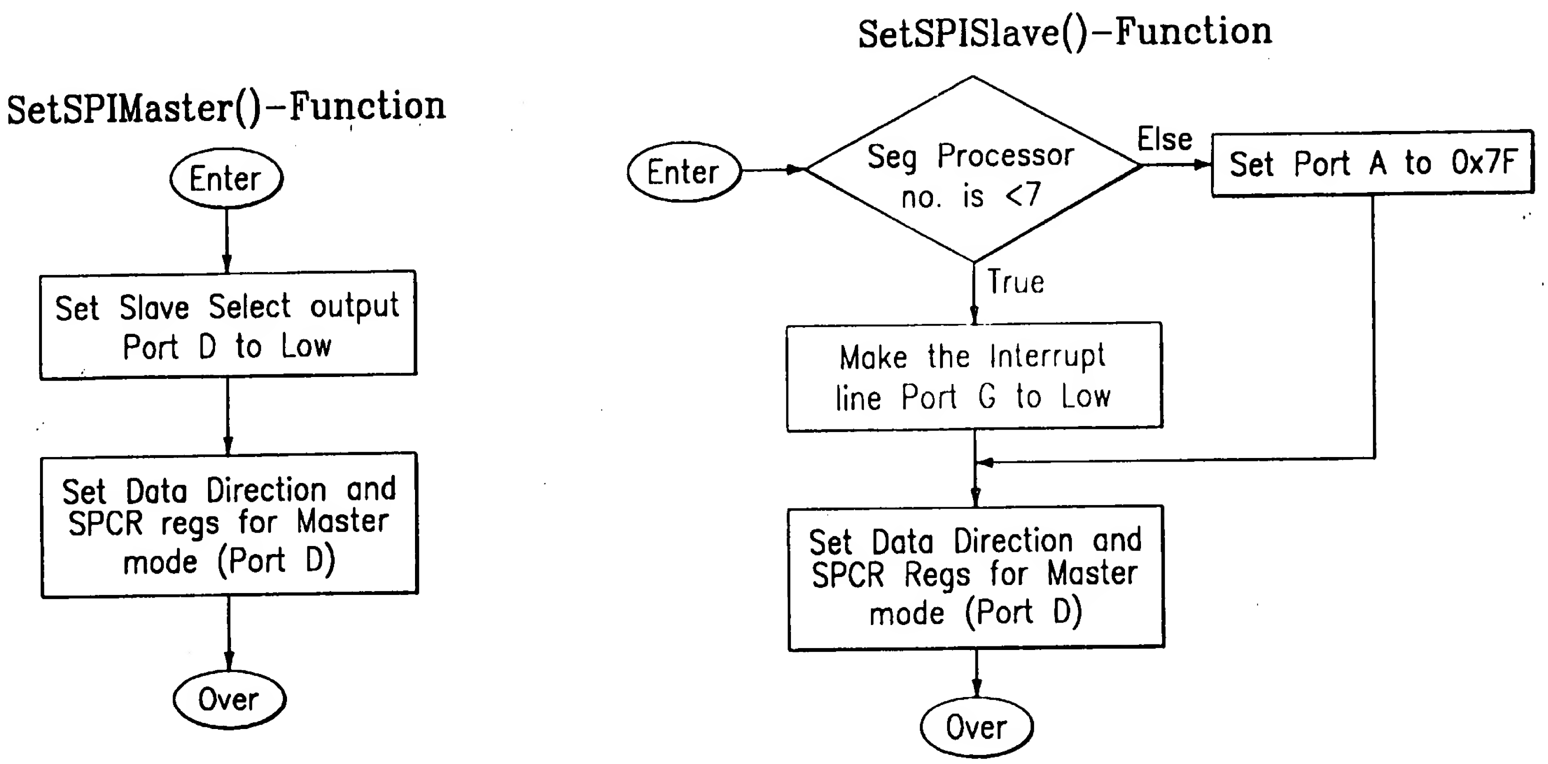

Enter
Read IPCState Flag
Check Flag== IPC Receive state?
True
Check for SPI receive Timeout Expired?
False
Set Seg Processor Error Condition to False Process the Response
Validate IPC CMD
IPC CMD is IPC Update Registers?
True
Read Device Address DynamicRegByteCount SetpointRegByteCount EventRegByteCount Values from IPC rx Buffer
Match Device Address by searching device Table
Update Device Table Registers for the selected Device
Break
Registers Updated are Dynamic Value Regs Setpoint Regs Event Regs
Break
From all breaks
Set SPI Request Flag to False Set IPC State Flag to IPC Transmit State
Return

FIG. 104



SendReqForProductId()–Function

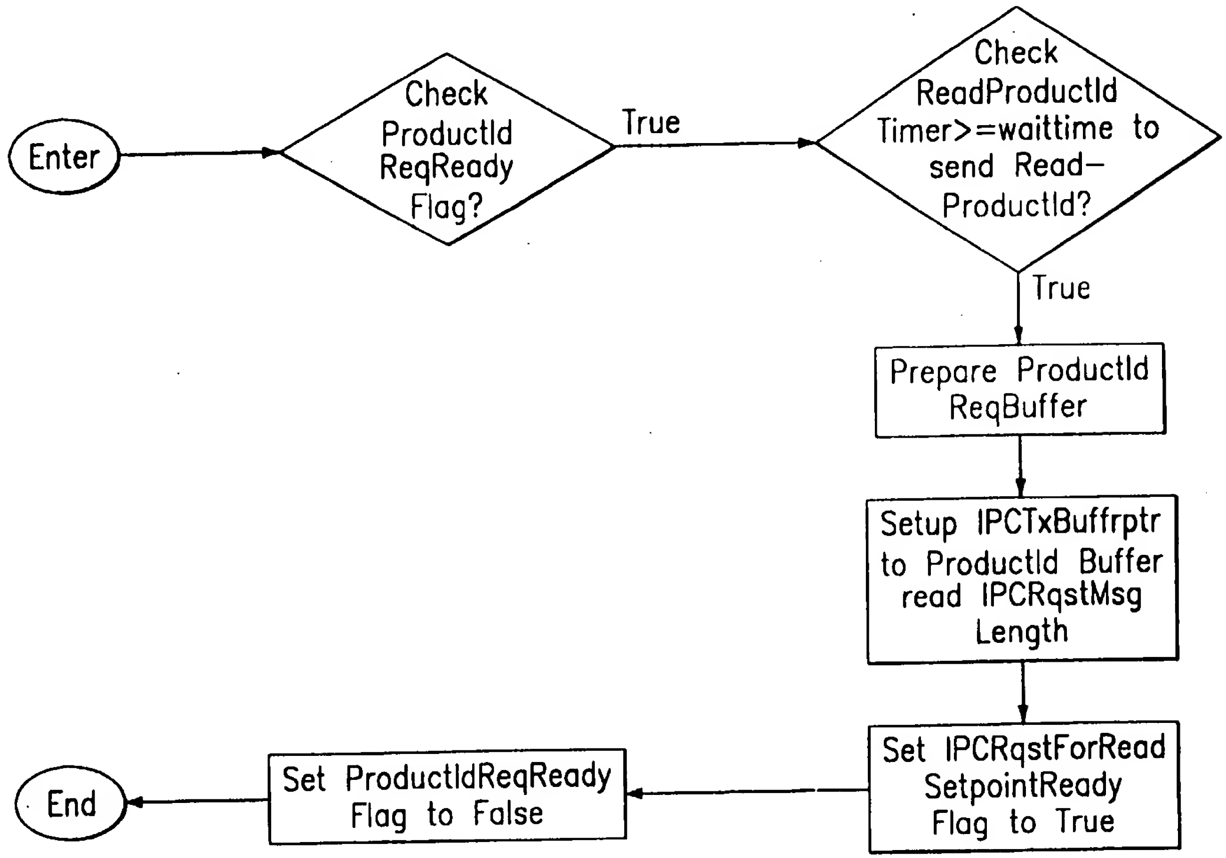


FIG. 106

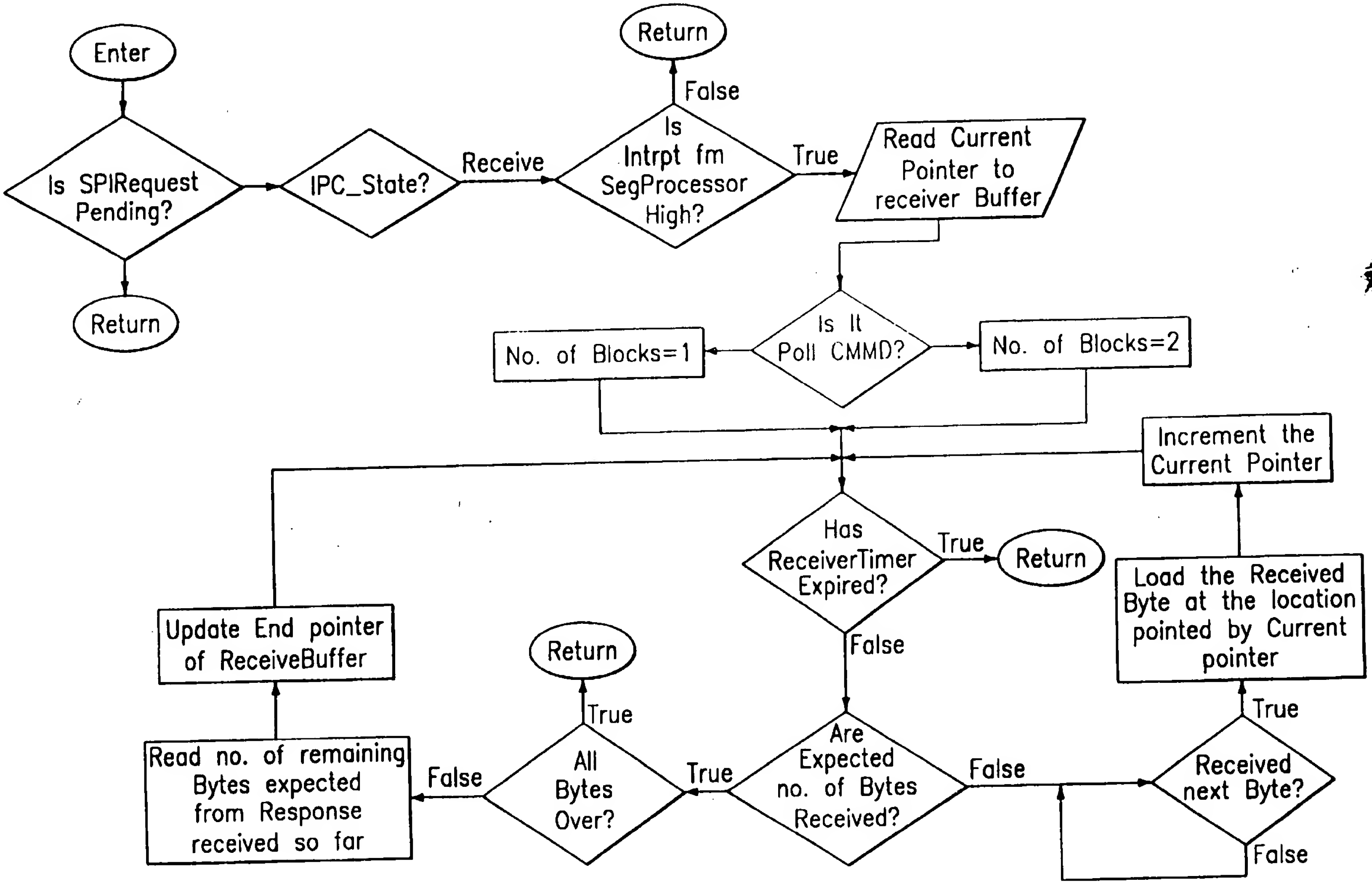

Enter
Is SPIRequest Pending?
Return
IPC_State?
Receive
Is Intrpt fm SegProcessor High?
False
Return
True
Read Current Pointer to receiver Buffer
Is It Poll CMMD?
No. of Blocks=1
No. of Blocks=2
Increment the Current Pointer
Has ReceiverTimer Expired?
True
Return
False
Load the Received Byte at the location pointed by Current pointer
True
Update End pointer of ReceiveBuffer
Return
True
All Bytes Over?
True
Are Expected no. of Bytes Received?
False
Received next Byte?
False
False
Read no. of remaining Bytes expected from Response received so far

FIG. 107

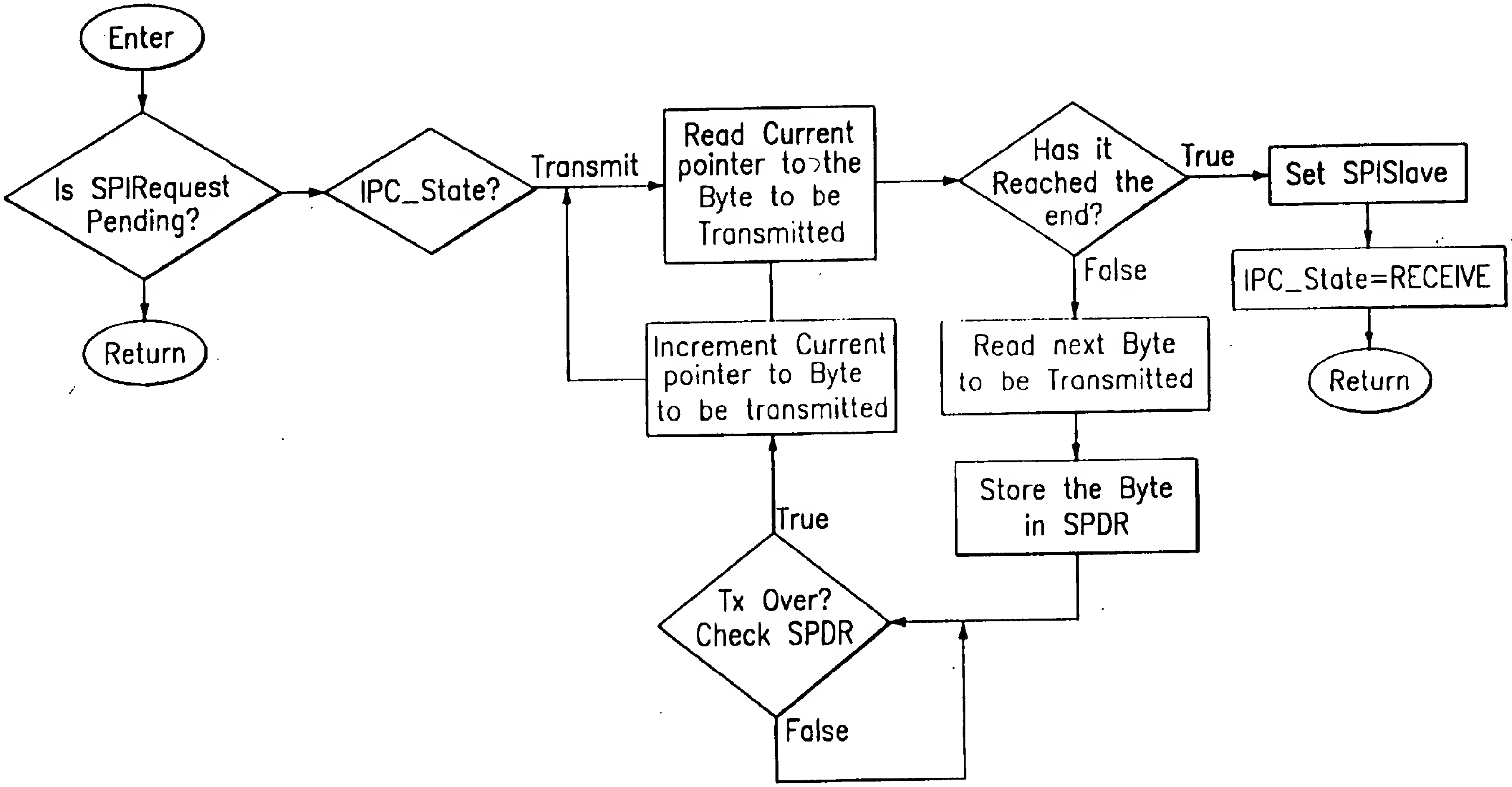

Enter
Is SPIRequest Pending?
Return
IPC_State?
Transmit
Read Current pointer to the Byte to be Transmitted
Has it Reached the end?
True
Set SPISlave
IPC_State=RECEIVE
Return
False
Read next Byte to be Transmitted
Store the Byte in SPDR
Tx Over? Check SPDR
True
False
Increment Current pointer to Byte to be transmitted

FIG. 108